

# Percolazione di siti su reticolo quadrato. Simulazione numerica

Carlo Meneghelli

Dicembre 2004

## Indice

<b>1</b>	<b>I modelli di percolazione</b>	<b>1</b>
<b>2</b>	<b>Piano del lavoro</b>	<b>2</b>
<b>3</b>	<b>Algoritmo di analisi—Cluster-counting</b>	<b>3</b>
3.1	L'algoritmo "etichetta" . . . . .	4
3.2	L'algoritmo di Hoshen-Kopelman . . . . .	4
3.2.1	Il problema del tempo . . . . .	5
<b>4</b>	<b>Come individuare la soglia</b>	<b>6</b>
4.1	Inroduzione di una nuova "manopola" . . . . .	9
4.2	$p_c$ dipende da $L$ ? . . . . .	9
4.3	Proposte alternative . . . . .	11
<b>5</b>	<b>Esponenti critici</b>	<b>11</b>
<b>6</b>	<b>Geometria del cluster infinito</b>	<b>13</b>
6.1	Giustificazione all'ipotesi di scaling . . . . .	13
<b>7</b>	<b>Conclusioni</b>	<b>15</b>
<b>A</b>	<b>Distanza di correlazione</b>	<b>15</b>
<b>B</b>	<b>Funzioni omogenee generalizzate</b>	<b>16</b>
<b>C</b>	<b>Grafici</b>	<b>18</b>
C.1	Esponenti critici . . . . .	18

## 1 I modelli di percolazione

Si consideri un reticolo di siti in cui i legami tra primi vicini possano essere presenti o assenti, nella **percolazione di legami** la relazione *essere collegato* è definita da:

- Ogni sito è collegato a se stesso;

- Se:  $\mathbf{r}_j$  è primo vicino di  $\mathbf{r}_i$  ed è presente il legame  $\langle ij \rangle$ , e  $\mathbf{r}_i$  è collegato al sito  $\mathbf{r}_k$ , allora  $\mathbf{r}_j$  e  $\mathbf{r}_k$  sono collegati;

Essere collegato è una relazione di equivalenza, i siti del reticolo si dividono in classi di equivalenza di siti collegati tra loro. Queste classi di equivalenza sono i cluster (ammassi).

Il problema della percolazione consiste nel determinare le proprietà geometriche dei cluster una volta note le distribuzioni statistiche dei legami.

Nella **percolazione di siti** ogni sito può essere occupato o non occupato: ogni sito occupato è collegato a se stesso, due siti diversi sono collegati se sono entrambi occupati e se sono “unibili” attraverso i primi vicini occupati.

Il problema standard della percolazione è quello in cui ogni sito (legame) è occupato (presente) con probabilità  $p$  indipendente dagli altri. La presenza (occupazione) del legame (sito) è una proprietà aleatoria.

**I modelli di percolazione** rappresentano in genere delle *miscele* (metallo/isolante, sani/malati, alberi/radure, aria/café, materiali porosi) e **descrivono dei fenomeni di trasporto** in esse (conduzione corrente, propagazione malattie, fuoco o liquido).

In questi sistemi si osserva, al crescere di  $p$ , che è la *concentrazione* di siti occupati (legami), la formazione di cluster di dimensioni maggiori. Questi cluster rappresentano i cammini lungo i quali si può propagare il liquido o l'informazione. Esiste un **valore critico  $p_c$** , la **soglia di percolazione**, al quale compare un cluster che connette i bordi del reticolo, nel limite termodinamico (in cui i lati del reticolo vanno all'infinito) questo cluster è infinito. Il valore di  $p_c$  dipende in generale dalle proprietà geometriche del reticolo. Il comportamento del sistema sopra e sotto la soglia è molto differente, es. conduttore e isolante, e la comparsa del cluster infinito (connessione a lungo raggio) è legata ad una sorta di transizione.

## 2 Piano del lavoro

Tra i tanti modelli di percolazione esistenti solo due si sanno risolvere analiticamente: reticolo di Bethe (struttura ad albero) e il caso unidimensionale (in cui percolazioni di siti e di legami di fatto coincidono). La simulazione numerica diventa lo strumento principale per lo studio di questi problemi. Questo è un esempio di soluzione numerica di problemi con il metodo **Monte Carlo**, si generano configurazioni (campioni) in accordo con una certa distribuzione (nel nostro caso in accordo con il valore di  $p$  assegnato) e sulle configurazioni generate si calcolano delle osservabili (nel nostro caso parametro d'ordine e suscettività). La bontà dei metodi Monte Carlo sta nel fatto che se si aggiungono campioni la statistica migliora e l'errore diminuisce.

In particolare verrà trattata la percolazione di siti su un reticolo quadrato (2d) in cui ogni sito può essere vuoto o occupato, sia  $p$  la probabilità che un sito sia occupato,  $1 - p$  sarà la probabilità che sia vuoto, il reticolo è *bernoulliano* ovvero ogni sito ha probabilità indipendente di essere vuoto o occupato. Due siti sono primi vicini se hanno un lato in comune, uno spigolo non è sufficiente. Un cluster è definito come un gruppo di siti occupati primi vicini (*lattice animals*), la sua taglia è il numero di siti occupati di cui è composto.

Il primo problema da analizzare è quello di **trovare un buon algoritmo che ci permetta di individuare i cluster**, verranno poi studiate certe quantità per

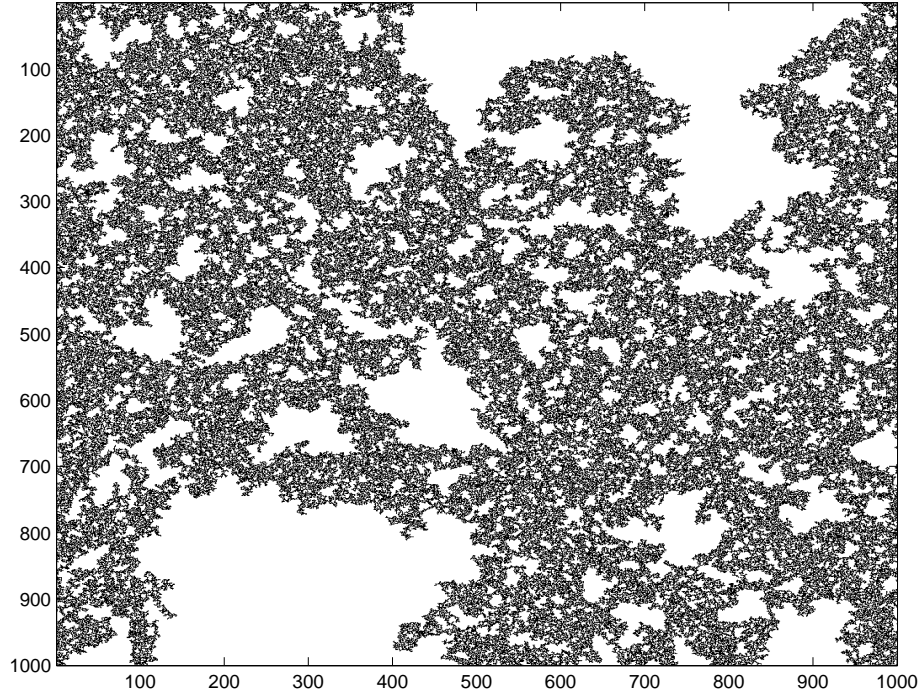


Figura 1: Spanning cluster,  $L = 1000$ ,  $p = p_c = 0.592746$ , sono state usate *periodic boundary condition*

individuare la soglia e il loro andamento vicino a  $p_c$  (esponenti critici). Alla fine per chiarire le idee verrà studiata la geometria del cluster infinito.

La questione è: come trattare un reticolo infinito avendo a disposizione solo reticoli finiti, bisognerà stare attenti ai *finite size effects*.

### 3 Algoritmo di analisi—Cluster-counting

Per analizzare i reticoli verranno proposti due algoritmi a confronto, naturalmente verrà utilizzato quello che risulterà più veloce per calcolare le quantità a cui si è interessati<sup>1</sup> (vedi Sez. 4).

**Creiamo il reticolo:** fissato un valore di  $p$  per ogni sito del reticolo viene estratto un numero random  $r$  (dal generatore di *Matlab*) uniformemente distribuito tra zero e uno, se  $r \leq p$  il sito è considerato pieno, in caso contrario vuoto<sup>2</sup>. Si avrà così una matrice di zeri e uni in cui gli zeri corrispondono a siti vuoti e gli uni a siti occupati.

<sup>1</sup>in particolare deve essere in grado di restituire le taglie di tutti i *cluster* presenti in un reticolo arbitrario.

<sup>2</sup>l'argomento *random number generator* non verrà approfondito, si ipotizza che il generatore usato sia buono per il problema affrontato, la bontà dei risultati potrà confermare l'ipotesi

### 3.1 L'algoritmo "etichetta"

Questo algoritmo parte da un sito occupato e identifica l'intero *cluster* a cui questo sito appartiene, nell'analisi **etichetta i siti già visitati per non visitarli più**. Immaginiamo di avere due strutture<sup>3</sup>, la prima, chiamata *cluster*, deve contenere gli indirizzi dei siti appartenenti al cluster in esame, l'altra, chiamata *visitato*, ci dice se del sito visitato sono stati controllati i primi vicini.

Si parte da un sito occupato  $x$  e si trovano i primi vicini, i loro indirizzi verranno impilati nella struttura *cluster*, in *visitato* verrà "annotato" che si sono trovati i vicini di  $x$ , a questo punto si visiteranno i vicini dei vicini appena impilati e così via fino a esaurire il cluster.

Alla fine la taglia del cluster sarà il numero di elementi nel vettore *cluster*.

Come l'algoritmo del prossimo paragrafo l'algoritmo "etichetta" visita solo i siti occupati—se non sono etichettati— e lo fa per colonne partendo dallo spigolo in alto a sinistra per arrivare a quello in basso a destra.

Si può dire fin da subito che questo algoritmo ha il seguente vantaggio: partendo da un sito occupato a caso restituisce la taglia del cluster a cui appartiene e lo identifica<sup>4</sup>. Mentre il prossimo algoritmo proposto è "costretto" a analizzare l'intero reticolo.

### 3.2 L'algoritmo di Hoshen-Kopelman

L'algoritmo di *Hoshen-Kopelman* (che d'ora in poi verrà chiamato algoritmo HK76, [HK76]) è un esempio di **cluster multiple labeling technique**. **Il reticolo viene visitato sito per sito per colonne partendo dallo spigolo in alto a sinistra per arrivare a quello in basso a destra**. Per spiegare meglio il suo funzionamento a linee generali è utile analizzare il piccolo reticolo in figura, i pallini neri indicano siti occupati.

●	●
○	○
○	●
●	●

**Quando si incontra un sito occupato che non è connesso ad altri siti occupati sopra o a sinistra inizia un nuovo *cluster* e viene assegnato al sito un nuovo *label***<sup>5</sup>, per l'assegnazione dei *label* si partirà dal valore uno. Invece, quando c'è un primo vicino sopra o a sinistra occupato (uno solo dei due) il sito in esame prende il valore del sito primo vicino occupato, analogamente se i suoi primi vicini sono entrambi occupati ma con lo stesso *label*. Dopo sette passi il nostro reticolo appare così:

1	1
	3
2	?

<sup>3</sup>si deve pensare ad una struttura tipo pila, mentre di fatto saranno utilizzati due vettori.

<sup>4</sup>può essere utile in casi come l'algoritmo di *Ising* a mosse globali

<sup>5</sup>numero che identifica i cluster, in linea di principio siti appartenenti a cluster diversi hanno *label* diversi

L'ottavo sito ha due primi vicini occupati con valori (*label*) differenti, quale viene scelto dei due? Si sceglie il minore ma bisogna fare attenzione perché in realtà il sito con valore 3 (anzi, tutto il "cluster 3") appartiene allo stesso cluster. Cambiare il valore di tutti i 3 comporterebbe un duro lavoro, soprattutto per grossi cluster. L'idea dell'algoritmo HK76 è di non assegnare nuovi valori ai siti (nell'esempio a quello identificato dal 3), ma prendere nota che 2 e 3 sono lo stesso cluster. In pratica questo viene fatto usando un vettore chiamato *Label of Label (LofL)* che contiene tutta l'informazione necessaria sui label dei cluster: per un *good label* (come 2 nel caso precedente) memorizza la taglia (momentanea) del cluster, per *bad label* (come 3 nel caso precedente) memorizza qual è il vero cluster label a cui questo label appartiene. Questa distinzione viene fatta attraverso il segno del numero intero contenuto nella componente in esame di LofL. Funziona in questo modo:

- $LofL(goodlabel) =$  taglia
- $LofL(badlabel) =$  indirizzo del label di riferimento<sup>6</sup>

Questo compito viene svolto da **HKclass** che preso in ingresso un label (e naturalmente *LofL*) restituisce il *good label* corrispondente. Di fatto HKclass si autorichiama finché il label non è positivo.

Alla fine l'algoritmo HK (a meno che non venga fatta una rilabelizzazione successiva) non garantisce che tutti i siti di un fissato cluster abbiano lo stesso valore (mentre siti di cluster diversi non possono avere lo stesso valore) ma restituisce in modo corretto le taglie dei cluster (unica quantità a cui siamo interessati per la nostra analisi).

**Osservazione:** la labelizzazione dei siti e la classificazione dei cluster possono essere svolti simultaneamente alla creazione del reticolo, vantaggi: si fa la simulazione di un grande reticolo senza doverlo memorizzare tutto. Dato che il reticolo viene visitato colonna per colonna e servono informazione solo riguardo ai primi vicini sopra e a sinistra rispetto al sito in esame, si possono tenere in memoria solo due righe alla volta (o più nel caso di condizioni periodiche).

Questo vantaggio è particolarmente utile in dimensioni maggiori di due, se si lavora in  $d$  dimensioni si memorizzano oggetti in  $d - 1$  dimensioni<sup>7</sup>.

### 3.2.1 Il problema del tempo

Per il problema in esame l'algoritmo HK76 risulta il più adatto per la sua velocità, infatti il tempo impiegato è proporzionale al numero di siti nel reticolo, ovvero a  $L^2$  (meglio di così non si può fare, si può solo lavorare sulla costante di proporzionalità). In un primo tempo la versione implementata dell'algoritmo (su *matlab*) non rispettava questa linearità nei tempi, ma presentava un rallentamento critico. Subito ho pensato che il problema venisse dalle troppe autochiamate di *Hkclass* al crescere di  $L$ , e che fosse quindi necessaria una *path compression*<sup>8</sup> e con un contatore ho verificato che il numero di chiamate a *Hkclass* aveva un andamento lineare (giustamente), rispetto a  $L^2$ . Cosa allora

<sup>6</sup>non necessariamente è il *good label*, possono essere necessari più passi per arrivare al valore corretto.

<sup>7</sup>Questo approccio è stato analizzato, ma nel caso di due dimensioni non esistono quasi problemi di memoria, la limitazione più grande è il tempo di analisi.

<sup>8</sup>riduzione del cammino per arrivare da un bad label al good label associato

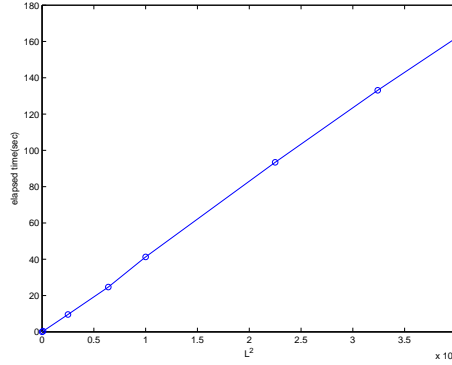


Figura 2: Il tempo impiegato per analizzare un reticolo é perfettamente lineare rispetto al numero di siti di cui il reticolo é composto (i valori sopra si riferiscono a *free boundary conditio* e  $p = p_c$ )

determinava il rallentamento critico? Problemi di memoria? Ho guardato se impiegava piú tempo a visitare la prima metà o la seconda ma il tempo era lo stesso. Alla fine, utilizzando il *profiler* di matlab, mi sono accorto che ogni volta che era necessario valutare una componente di LofL il calcolatore sprecava un sacco di tempo. Sono quindi arrivato alla conclusione che LofL definito *sparse* rallenta criticamente, definito *full* fa sí che landamento dei tempi sia come ci si aspetta. Ho ottenuto cosí un tempo lineare col numero di siti (a fissata  $p$ ).

## 4 Come individuare la soglia

In prima analisi per individuare  $p_c$  verranno analizzate due quantità, seguendo di fatto l'articolo di Hoshen-Kopelman [HK76], che sono la dimensione media dei cluster e la probabilità che un sito appartenga al cluster infinito.

La dimensione media dei cluster  $S$  (**mean cluster size**) é utile per individuare  $p_c$  poiché alla soglia presenta un massimo piuttosto netto<sup>9</sup>. Non é difficile rendersi conto che la dimensione media dei cluster può essere cosí calcolata:

$$(4.1) \quad S = \frac{\sum s^2 n_s}{\sum s n_s}$$

Dove  $n_s$  é il numero di cluster di taglia  $s$  per sito e  $s$  la taglia, dalla somma si esclude il cluster infinito (naturalmente solo quando presente, ovvero sopra la soglia) sia al denominatore che al numeratore. In realtà verrà analizzata una quantità, leggermente diversa da questa, chiamata **reduced average cluster size**(RACS).

$$(4.2) \quad RACS = \frac{\sum s^2 n_s - s_{max}}{\sum s n_s}$$

mah! credo ci voglia  
un quadrato ad smax

Questa quantità differisce dalla precedente poiché il cluster infinito é sostituito dal cluster massimo (sia sopra che sotto la soglia) e poiché il contributo del

---

<sup>9</sup>All'aumentare di  $p$  infatti si formeranno cluster di dimensioni sempre maggiore, che rimarranno finiti per  $p < p_c$ ,  $S$  aumenterà; superata la soglia il contributo del cluster infinito viene escuso dalla media e  $S$  diminuisce.

cluster massimo é escluso solo al numeratore<sup>10</sup>. Sostituire il cluster infinito con il massimo é abbastanza naturale visto che si ha a che fare solo con reticoli finiti e che non si può sapere a priori (almeno in prima analisi) se il cluster massimo coincide o meno con uno *spanning cluster* (i.e. cluster che ha “percolato”). La scelta di contare il contributo dal cluster massimo al denominatore ha lo scopo di rendere la decrescita di RACS molto rapida dopo la soglia.

Esiste una corrispondenza tra la dimensione media dei cluster per i problemi di percolazione e la *suscettività* nei modelli magnetici (*Ising*), entrambe tendono a divergere alla soglia.

La seconda quantità calcolata é la probabilità che un sito appartenga al cluster infinito  $P$  (**the strenght of the infinite network**), corrisponde al *parametro d'ordine* del problema ed é definita in questo modo:

$$(4.3) \quad P(p) = \frac{s_{max}}{L^2}$$

Anche in questo caso verrà utilizzata una quantità leggermente diversa ovvero  $\frac{s_{max}}{\sum s_n}$  che rappresenta la probabilità che un sito occupato (non un sito arbitrario) appartenga al cluster massimo<sup>11</sup>. Nel limite termodinamico  $P$  é zero per  $p < p_c$  e tende con un certo andamento (decisamente rapido vicino a  $p_c$  poi più dolce) a uno. Si può definire  $p_c$  come il punto in cui  $P$  diventa per la prima volta non nullo.

Per chiarire le idee é utile accennare anche alla **probabilità di percolazione**  $\Pi$  che per un reticolo infinito corrisponde esattamente alla funzione a gradino (vale zero per  $p < p_c$  e uno per  $p > p_c$ ) mentre per reticoli finiti ha un andamento meno netto poiché esiste una probabilità di trovare uno *spanning cluster* sotto la soglia e analogamente di non trovarne uno sopra la soglia. Su questa base é fondato un altro metodo per individuare  $p_c$  a cui si farà un accenno in seguito, che consente un'individuazione molto accurata ma che richiede un numero molto grande di reticoli analizzati.

A gestire l'analisi di  $S$  e  $P$  é la funzione **simulperc**<sup>12</sup>, che, chiedendo in ingresso la taglia del reticolo  $L$ , le concentrazioni a cui analizzarlo e il numero di campioni da analizzare per ogni  $p$ , restituisce il valor medio delle due quantità in analisi. É possibile scegliere tra condizioni al bordo libere (*free boundary condition*) e periodiche<sup>13</sup> (*periodic boundary condition*). Ci si aspetta che l'introduzione di condizioni al bordo periodiche porti ad un andamento più liscio delle quantità che analizziamo, di fatto condizioni periodiche rendono più netto l'andamento di  $\Pi$  per  $L$  finito. L'ipotesi é confermata dall'analisi dei risultati, quindi si sceglie di lavorare sempre<sup>14</sup> con condizioni al bordo periodiche. In Fig. 3 sono mostrate le due quantità, l'andamento qualitativo rispecchia molto bene quello che ci si aspetta.

---

<sup>10</sup>risulta chiaro che di fatto al denominatore compare il numero di siti occupati diviso per il numero totale di siti

<sup>11</sup>a denominatore si conta anche il cluster infinito quindi di fatto vi compare il numero di siti occupati.

<sup>12</sup>ve ne sono varie versioni ma a grandi linee seguono lo stesso schema.

<sup>13</sup>In questo caso il reticolo assume una geometria di tipo cilindrico, esiste la possibilità di introdurre una geometria tipo *toro*, ma non é stata analizzata.

<sup>14</sup>a parte che per gli esponenti caratteristici

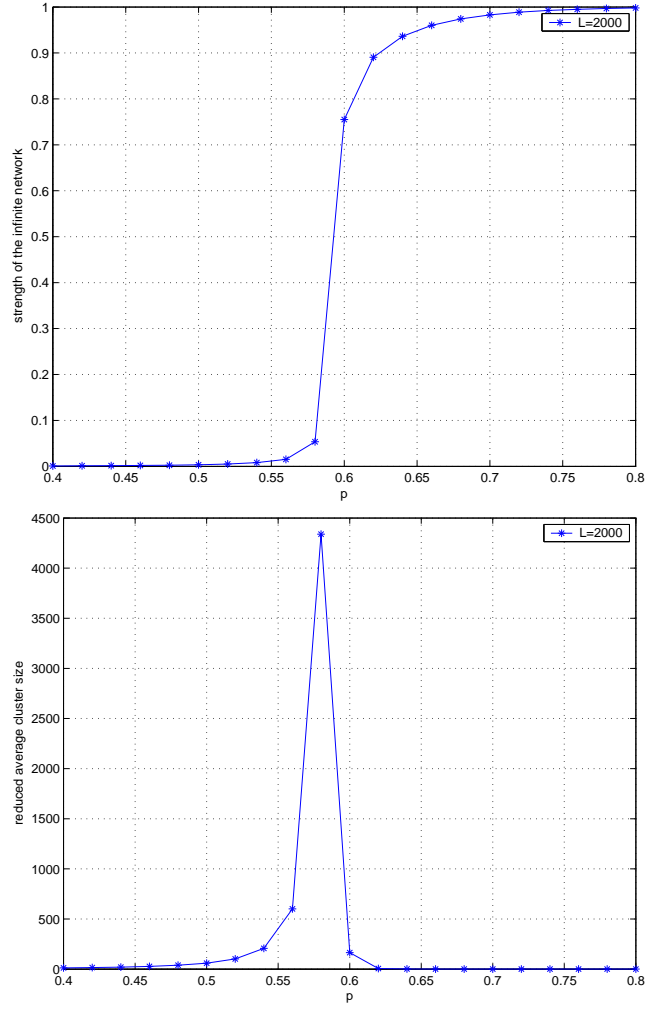


Figura 3: Andamento qualitativo di  $P$  e  $RACS$ , sono state eseguite 10 iterazioni per punto, sono state usate *periodic boundary condition*.



## 4.1 Introduzione di una nuova “manopola”

In un secondo momento é stato introdotto un controllo sui reticoli generati: solo se la differenza tra la concentrazione effettiva e quella richiesta ( $p$ ) é minore di una certa tolleranza il reticolo viene accettato e analizzato<sup>15</sup>. Ora si hanno a disposizione due manopole, la tolleranza e il numero di iterazioni (oltre naturalmente a  $L$  e uno “switch” per le condizioni al bordo), ci si aspetta che la tolleranza diminuisca in una certa misura le fluttuazioni<sup>16</sup>.

Si può valutare teoricamente, senza troppe difficoltà, la probabilità che un reticolo generato venga rifiutato. Ci si aspetta che, a fissata tolleranza, la probabilità di rigetto diminuisca al crescere delle dimensioni del reticolo e che questa dipenda anche da  $p$  (sarà massima per  $p = \frac{1}{2}$ ). Bisogna fare riferimento alla *distribuzione binomiale*. Su un grande numero di reticoli di taglia  $L$  generati in accordo a  $p$ , ci si aspetta che in media la concentrazione sia quella assegnata e la varianza sia del tipo  $\frac{p(1-p)}{L^2}$ . La tolleranza, confrontata con la varianza della distribuzione, fornisce un’idea piuttosto buona della probabilità di accettazione (anche perché si può pensare di avere a che fare con una distribuzione gaussiana senza sbagliare di troppo).

Bisogna inoltre sottolineare che anche quando la probabilità di rigetto é abbastanza alta (0.9 ad esempio), il tempo impiegato per generare campioni e accettarne uno é minimo rispetto a quello impiegato per analizzarlo (nel senso di contarne i cluster). Naturalmente se si potesse analizzare un numero molto grande di “campioni” non sarebbe necessario introdurre la manopola della tolleranza, ma mentre aumentare il numero di campioni fa aumentare il tempo di analisi (con un errore che va come  $\frac{1}{\sqrt{N}}$ ) introdurre una tolleranza non cambia i tempi di analisi.

In Fig. 4 vengono confrontati i risultati ottenuti con e senza tolleranza (in entrambi i casi vengono analizzati 50 campioni per ogni  $p$  e condizioni periodiche al contorno), la bontà del metodo qui introdotto é confermata dal confronto con [S99] in cui sono generati  $10^6$  per ogni  $p$  campioni (e manca il controllo sui reticoli generati).

## 4.2 $p_c$ dipende da $L$ ?

Nel tentativo di individuare la soglia nel modo più preciso possibile ci si chiede presto se  $p_c$  dipenda o meno da  $L$ , osservando la Fig. 4 sembra dipendervi<sup>17</sup>, esistono però metodi molto precisi per individuare la soglia che si basano sul fatto che  $p_c$  non dipende da  $L$ <sup>18</sup>. Il fatto é che, mentre  $p_c(\infty)$  é ben definita, esistono tanti modi per definire  $p_c(L)$ , di conseguenza l’andamento di  $p_c(L)$  può essere diverso. La soglia in dimensioni finite può essere definita come il massimo di  $S$ , oppure come il punto in cui  $P$  ha la massima pendenza, oppure come il valor medio di  $p$  pesato sulla probabilità che il cluster infinito compaia per la prima volta, oppure come il valore di  $p$  per cui la probabilità di percolazione vale  $\frac{1}{2}$  (o  $\frac{1}{3}$ , o  $\frac{2}{3}$ ) qualunque di queste sia la definizione (ce ne possono essere anche

<sup>15</sup>HK originale non poteva farlo perché non memorizzava l’intero reticolo.

<sup>16</sup>in generale per i metodi di soluzione numerica vale questo: é importante sapere che il metodo converge alla soluzione, ma é molto importante far sí che vi converga il più velocemente possibile.

<sup>17</sup>Se l’andamento in figura é poco convincente sicuramente quello di [S99] lo é di più

<sup>18</sup>vedi [W04]

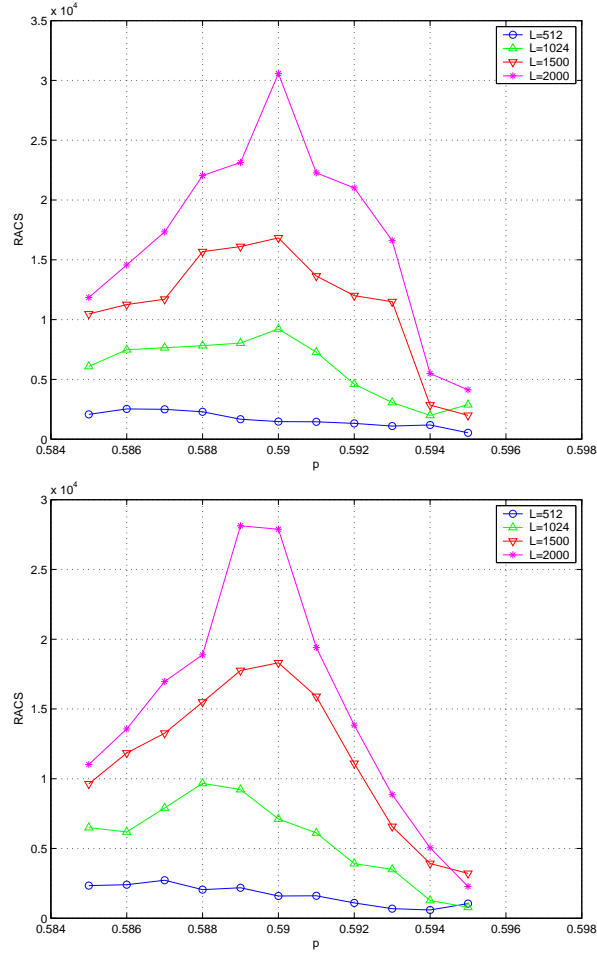


Figura 4: Per entrambi sono state usate *periodic boundary conditio*, 50 iterazioni per ogni punto, nel primo non é stata introdotta una tolleranza, nel secondo  $t = 10^{-4}$ . La tolleranza é la stessa per tutti, ma sicuramente é meno efficace per  $L$  grandi, sarebbe stato piú intelligente cambiare la tolleranza in base a  $L$ .

altre) vale un andamento di questo tipo:

$$(4.4) \quad p_c(L) - p_c(\infty) \propto L^{-\frac{1}{\nu}}$$

Siamo nell'ambito della **teoria di scaling** per l'analisi dei *finite size effects* (App. B), quindi la seguente relazione è valida per  $L$  abbastanza grandi e vicino alla soglia. La costante di proporzionalità dipende da come è definito  $p_c(L)$ , in particolare può essere nulla, in tal caso  $p_c$  non dipenderà da  $L$ . L'equazione suggerisce come calcolare  $p_c(\infty)$ , estrapolandola dall'andamento di  $p_c(L)$  ma bisogna aver determinato  $p_c(L)$  in modo molto preciso.

Il lavoro svolto ha avuto più lo scopo di valutare l'andamento qualitativo di  $S$  e  $P$  al variare di  $p$  e di  $L$  (anche se comunque con una certa precisione) che di calcolare  $p_c$  in modo preciso.

### 4.3 Proposte alternative

Esistono vari modi per individuare  $p_c$  in modo più preciso, qui di seguito ne vengono brevemente enunciati tre:

- fissare un reticolo, variare<sup>19</sup>  $p$ , individuare a quale  $p$  “percola” (deve connettere sopra e sotto) la media dei  $p$  individuati da  $p_c$ .
- metodo degli istogrammi<sup>20</sup>: si analizza direttamente la probabilità di percolazione ( $C = 1$  se il reticolo ha percolato, zero altrimenti, la media dei  $C$  dà un'idea di  $\Pi$ )
- metodo identico a quello usato in questo lavoro, ma, nel valutare  $S$  e  $P$ ,  $s_{max}$  è diverso da zero solo se il cluster massimo è percolante.

Gli ultimi due approcci sono stati provati, per il secondo servono troppi campioni (insorgono problemi di tempo) per avere risultati soddisfacenti, il terzo non comporta un miglioramento decisivo, mentre porta a un raddoppio dei tempi impiegati per analizzare i reticoli—rispetto alla tecnica generalmente usata—poiché li deve rilabellare.

## 5 Esponenti critici

Vicino alla soglia di percolazione i parametri del sistema (quantità di interesse) seguono andamenti descritti da leggi di potenza i cui esponenti sono chiamati **esponenti critici**<sup>21</sup>.

In particolare —nel limite termodinamico—  $P_\infty$  e  $S$  seguono un andamento di questo tipo:

$$(5.1) \quad P \propto (p - p_c)^\beta \quad (p > p_c)$$

<sup>19</sup> è consigliato un metodo “dicotomico” per risparmiare passi

<sup>20</sup> vedi [W04] e [H94]

<sup>21</sup> L'importanza degli esponenti critici è evidenziata dall'ipotesi di universalità: le transizioni di fase possono essere distinte in un piccolo numero di classi di universalità caratterizzate dalla dimensione del sistema e del parametro d'ordine, e all'interno di ogni classe gli esponenti sono gli stessi.

$$(5.2) \quad S \propto |p - p_c|^{-\gamma}$$

Dove<sup>22</sup>:

$$(5.3) \quad P_\infty = \frac{s_{max}}{L^2} \quad S = \sum (s^2 n_s)$$

In questo caso  $\beta$ ,  $\gamma$  e  $\nu$  sono gli esponenti critici. Poiché le simulazioni avvengono su reticoli finiti (non siamo al limite termodinamico), gli andamenti sopra descritti —vicino a  $p_c$ — non dipenderanno solo dalla concentrazione  $p$  ma anche dalla taglia del reticolo. Per stimare le dimensioni frattali e gli esponenti critici si farà riferimento alla finite size scaling analysis<sup>23</sup>. Assumendo che solo la distanza di correlazione  $\xi$  domini l'andamento critico si introducono le funzioni di scala (*scaling function*)  $G$  e  $F$ .

$$(5.4) \quad P(L, p) = L^{-\frac{\beta}{\nu}} G((p - p_c(L))L^{\frac{1}{\nu}})$$

$$(5.5) \quad S(L, p) = L^{\frac{\gamma}{\nu}} F((p - p_c(L))L^{\frac{1}{\nu}})$$

Valida per  $P$  vicino alla soglia e  $L$  grandi.

Inoltre a  $p_c$  la massa del cluster infinito (Sez. 6) dello stesso ordine di  $L^{D_f}$ :

$$(5.6) \quad V = PL^d \propto L^{D_f}$$

Dove  $d$  sono le dimensioni euclidee e  $D_f$  le dimensioni frattali<sup>24</sup>. In questo caso di fatto non verranno determinati  $\beta$  e  $\gamma$  ma gli esponenti normalizzati. Per ricavare tutti gli esponenti critici ci sarebbe bisogno di misurare  $\nu$ , la strada per farlo viene direttamente dall'Eq. 4.4 il che richiede una misura molto precisa di  $p_c(L)$  [S99].

Naturalmente se si analizzano  $S$  e  $P_\infty$  a  $p_c(L)$  per vari  $L$  la pendenza della retta ottenuta dal grafico in scala logaritmica di  $P$  (o  $S$ ) rispetto a  $L$  dá gli esponenti normalizzati.

I risultati dell'analisi sono riportati in App. B con buon accordo con i valori tabulati degli esponenti. Ogni reticolo dovrebbe essere messo al "suo"  $p_c$ ,  $p_c$  non è stato calcolato in modo così preciso da poterlo fare e quindi è stato usato per tutti lo stesso valore di  $p_c$  ovvero quello tabulato:  $p_c = 0.592746$ <sup>25</sup>.

Inserendo la Eq. 5.4 nella Eq. 5.6 si può verificare che tra gli esponenti vale la seguente relazione<sup>26</sup>:

$$(5.7) \quad D_f = d - \frac{\beta}{\nu}$$

---

<sup>22</sup>nota al denominatore di  $S$  non compare nulla, di fatto dovrebbe esserci  $\sum n_s s$  che non è altro che  $p$ , componente irrilevante ai fini di individuare l'andamento a potenza.

<sup>23</sup>l'ipotesi di *scaling* viene presa in eredità dalle transizioni di fase del secondo ordine nei sistemi magnetici.

<sup>24</sup>Senza soffermarci troppo sul significato di dimensioni frattali, possiamo dire brevemente (e naturalmente in modo non molto corretto) che sono esponenti non interi che mettono in relazione le dimensioni lineari di un sistema con il suo volume.

<sup>25</sup>questa scelta può essere anche giustificata teoricamente ricordando che  $(p_c(L) - p_c(\infty))L^{\frac{1}{\nu}} = const_1$  e quindi anche  $F(const_1) = const_2$  è costante

<sup>26</sup>questa relazione è naturalmente verificata dagli esponenti che vengono calcolati con questa tecnica, le misure fatte di  $D_f$  e  $\frac{\beta}{\nu}$  non sono indipendenti

## 6 Geometria del cluster infinito

Lo scopo di questo paragrafo é analizzare la geometria del *cluster infinito* sopra la soglia<sup>27</sup>, quest'analisi é utile per capire il significato della distanza di correlazione e acquisire consapevolezza rispetto alla *teoria di scaling*.

Vicino alla soglia di percolazione su scale grandi rispetto a  $\xi$  lo *spanning cluster* riempie lo spazio in modo omogeneo, mentre é estremamente ramificato su scale minori di  $\xi$ , il cluster infinito (nel limite termodinamico) a  $p_c$  ( $\xi \rightarrow \infty$ ) é un **frattale**<sup>28</sup>.

Un modo per studiare la geometria dello *spanning cluster* é il seguente: si sceglie un punto sul reticolo a  $d$  dimensioni, lo si circonda con un *box* di dimensioni lineari pari a  $l$  e si considera la massa del cluster entro il box,  $\mu(l)$  e la *densità*  $\rho \equiv \frac{\mu(l)}{l^d}$ . A fissata  $p > p_c$  si generano reticoli di dimensioni  $L$  (inserendo una tolleranza come da Sez. 4.1 come punto di partenza si sceglie naturalmente il centro del reticolo, solo se il sito al centro del reticolo appartiene allo *spanning cluster* si procede a calcolare  $\rho(l)$ ).

Il grafico  $\ln(\rho)$  vs  $\ln(l)$  Fig. 5 presenta, all'aumentare di  $l$ , una prima parte rettilinea e successivamente un *plateau*, per alti valori di  $l$  si notano effetti di bordo. Per lunghezze minori di  $\xi$  (che viene identificata con l'intersezione tra le due rette) la densità non é uniforme e  $\mu \propto l^{D_f}$ , mentre per lunghezze maggiori di  $\xi$  la densità é costante e quindi  $\mu \propto \text{const } l^d$ . Dall'Eq. 5.3 é facile convincersi che la costante coincide con  $P_\infty$ , si ottiene il seguente andamento<sup>29</sup>:  $\mu(l) \propto Pl^d \propto \xi^{-\frac{d}{\nu}} l^d$ , per  $l > \xi$  (naturalmente vicino alla soglia).

Si può concludere che se si “guarda” il reticolo su scale minori di  $\xi$  si può ignorare il fatto di avere a che fare con uno  $\xi$  finito, il comportamento che si osserva é lo stesso che si osserva quando  $\xi$  é infinito (ovvero alla soglia).

$\xi$  rappresenta una lunghezza chiamata *crossover length*, per sottolineare come cambiano le cose passando da  $l < \xi$  a  $l > \xi$ ; in particolare se ci fissiamo a  $l = \xi$  possiamo valutare la massa del cluster infinito in due modi, dal confronto delle due quantità avremo una giustificazione dell'Eq. 5.7 senza introdurre direttamente l'ipotesi di scaling.

$$(6.1) \quad P_\infty l^d \propto l^{D_f} \quad \text{per } l = \xi \propto (p - p_c)^{-\nu}$$

Da cui segue:

$$(6.2) \quad D_f = d - \frac{\beta}{\nu}$$

Analizzando risultati del tipo di quelli riportati in Fig. 5 per varie  $p$  (maggiori di  $p_c$  ma comunque vicino alla soglia) si possono ottenere misure indipendenti per  $\beta$ ,  $D_f$  e  $\nu$  rendendo possibile una verifica diretta dell'Eq. 6.2 [K83].

### 6.1 Giustificazione all'ipotesi di scaling

Alla luce delle osservazioni appena fatte si può analizzare l'andamento di  $P_\infty$  come funzione di  $L$  e di  $p$ , verranno ritrovate equazioni già viste (Eq. 5.4, Eq. 5.5),

<sup>27</sup>il cluster infinito non é presente sotto la soglia

<sup>28</sup>associato al concetto di dimensioni frattali (Mandelbrot 1977)vi é quello di autosimilarità (*self similarity*)

<sup>29</sup>si sono utilizzate le relazioni:  $\xi \propto |p - p_c|^\nu$ ,  $P \propto |p - p_c|^\beta$ .

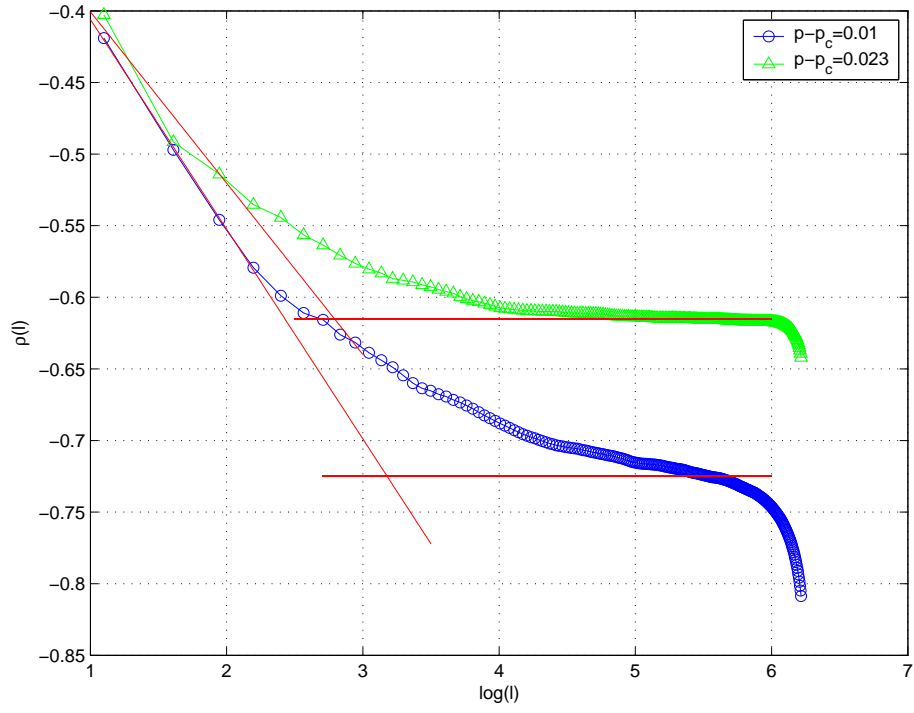


Figura 5:  $L=500$ , sono stati analizzati 500 campioni, *free boundary condition*, *tolleranza* =  $10^{-4}$

il risultato verrà generalizzato a generiche funzioni  $\Lambda(L, p)$  (naturalmente tutto ciò é valido per  $L$  abbastanza grandi e vicino alla soglia). Dal paragrafo precedente é banale stabilire l'andamento di  $P_\infty(L, \xi)$ :

$$(6.3) \quad P_\infty \propto (p - p_c)^\beta = \xi^{-\frac{\beta}{\nu}} \quad \text{per } L \gg \xi$$

$$(6.4) \quad P_\infty \propto L^{D_f} L^{-d} = L^{-\frac{\beta}{\nu}} \quad \text{per } L \ll \xi$$

Da cui:

$$(6.5) \quad P_\infty(L, \xi) = \xi^{-\frac{\beta}{\nu}} f_1\left(\frac{L}{\xi}\right)$$

oppure in un'altra forma:

$$(6.6) \quad P_\infty(L, p) = (p - p_c)^\beta f_2((p - p_c)L^{\frac{1}{\nu}})$$

$$(6.7) \quad \text{oppure } P_\infty(L, p) = L^{-\frac{\beta}{\nu}} f_3((p - p_c)L^{\frac{1}{\nu}})$$

Che non é altro che l'Eq. 5.5.

Se una quantità  $\Lambda$  vá come  $|p - p_c|^{-\lambda}$  per  $L > \xi$  ci aspettiamo per analogia che valga:

$$(6.8) \quad \Lambda(L, p) = L^{\frac{\lambda}{\nu}} \Phi((p - p_c)L^{\frac{1}{\nu}})$$

## 7 Conclusioni

I risultati ottenuti sono soddisfacenti, si é partiti da un tentativo—di fatto non riuscito— di individuare  $p_c$  con grande precisione per poi analizzare l'andamento di parametri importanti e la geometria del sistema. La percolazione é uno degli esempi piú semplici di transizione di fase ed ha tanti aspetti in comune con il modello di *Ising*<sup>30</sup>, anche se il problema della percolazione, a differenza di Ising, non ha un'hamiltoniana, il sistema é completamente random, non ci sono *memory effects*.

## A Distanza di correlazione

Definiamo il raggio di un cluster in questo modo:

$$(A.1) \quad R_s^2 = \sum \frac{|r_i - r_0|^2}{s}$$

dove  $r_0 = \sum \frac{r_i}{s}$

---

<sup>30</sup>il modello di Ising può essere mappato in un modello di percolazione generalizzato

Possiamo riscrivere  $R_s$  in termini della distanza media tra due siti di un cluster:

$$(A.2) \quad 2R_s^2 = \sum_{i,j} \frac{|r_i - r_j|^2}{s^2}$$

Definiamo la funzione di correlazione, o connettività,  $g(r)$  come la probabilità che un sito a distanza  $r$  da un altro appartenga allo stesso cluster,  $\sum g(r)$  risulterà il numero medio di siti ai quali un sito occupato è connesso. Questo numero è uguale a  $\sum \frac{s^2 n_s}{p}$ , dato che  $\frac{n_s}{p}$  è la probabilità che un sito occupato appartenga ad un s-cluster, quindi:  $\sum s^2 n_s^2 = p \sum g(r)$ .

Definiamo la distanza di correlazione (*correlation length*) come una qualche distanza media tra due siti appartenenti allo stesso cluster:

$$(A.3) \quad \xi^2 = \frac{\sum r^2 g(r)}{\sum g(r)}$$

Poiché per un dato cluster,  $2R_s^2$  è la distanza quadratica media tra due siti del cluster e poiché un sito appartiene con probabilità  $n_s s$  ad un s-cluster e quindi con la stessa probabilità è connesso a  $s$  siti, la distanza di correlazione può essere ottenuta da una media su  $2R_s^2$ :

$$(A.4) \quad \xi^2 = \frac{2 \sum R_s^2 s^2 n_s}{\sum s^2 n_s}$$

A parte fattori numerici la distanza di correlazione è il raggio dei cluster che danno maggior contributo a  $S$  vicino a  $p_c$ . Ci si aspetta quindi che  $\xi$  diverga alla soglia come:

$$(A.5) \quad \xi \propto |p - p_c|^{-\nu}$$

In relazione all'ipotesi di scaling si assume che una e una sola lunghezza, ovvero  $\xi$  domini l'andamento critico.

## B Funzioni omogenee generalizzate

L'ipotesi di scaling consiste nel considerare quantità come  $P_\infty(L, p)$  e  $S(L, p)$ , asintoticamente<sup>31</sup> funzioni omogenee generalizzate dei loro argomenti<sup>32</sup>. Si dice che  $f(x_1, \dots, x_n)$  è una funzione omogenea generalizzata se esistono i numeri  $a, a_1, \dots, a_n$  tali che, per ogni valore di  $\lambda > 0$ , si ha  $f(\lambda_1^a x_1, \dots, \lambda_n^a x_n) = \lambda^a f(x_1, \dots, x_n)$ . Per fare un esempio pratico si può prendere  $P_\infty(L, p)$ , posto  $x = \frac{(p-p_c)}{p_c}$  e  $y = \frac{1}{L}$ ,  $P_\infty$  può essere pensato come funzione di  $x$  e di  $y$ , nel limite per  $x$  e  $y$  che tendono a zero si ha:

$$(B.1) \quad P_\infty(\lambda^a x, \lambda^b y) = \lambda^c P_\infty(x, y)$$

<sup>31</sup>vicino alla soglia e per  $L$  grandi

<sup>32</sup>analogamente si può formulare l'ipotesi di scaling per  $n_s(p)$  pensato come funzione di  $s$  e di  $p$ , sarà valida vicino alla soglia per  $s$  grandi.



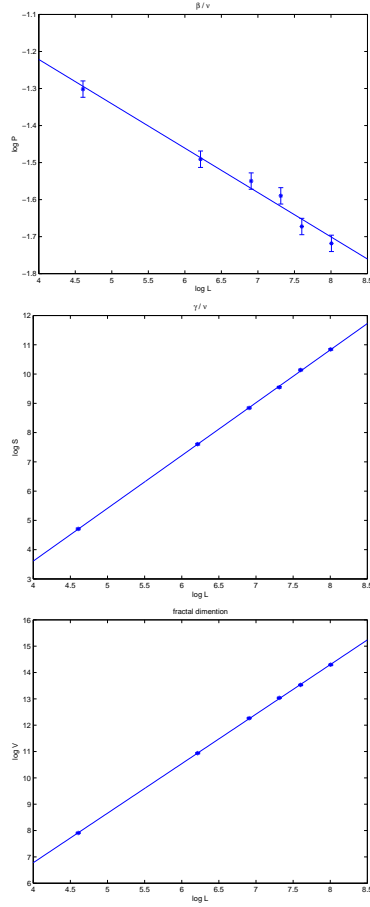


Figura 6:  $L=100,500,1000,1500,2000$ . *free boundary condition*, *tolleranza* =  $10^{-3}$ , 100 iterazioni per punto. Dove non ci sono le barre d'errore significa che sono troppo piccole e non si vedono. Risulta:  $D_f = 1.88 \pm 0.02$ ,  $\frac{\beta}{\nu} = 0.12 \pm 0.02$ ,  $\frac{\gamma}{\nu}$  da confrontare con i valori tabulati:  $D_f = 1.896$ ,  $\frac{\beta}{\nu} = 0.104$ ,  $\frac{\gamma}{\nu} = 1.792$ .

Ponendo  $\lambda^b y = 1$  si ottiene:  $P_\infty(x, y) = y^{\frac{\gamma}{b}} P_\infty(xy^{-\frac{\gamma}{b}}, 1)$  cioè:

$$(B.2) \quad P_\infty(L, p) = L^{-\frac{\beta}{\nu}} P_\infty((p - p_c)L^{\frac{1}{\nu}})$$

Con opportuno aggiustamento degli esponenti per ritrovare esponenti già noti (l'andamento rimane a quello a prescindere dal nome che diamo agli esponenti).

## C Grafici

### C.1 Esponenti critici

#### Riferimenti bibliografici

- [HK76] Hoshen J.,Kopelman R., Phys.Rev.B, **14**, 8, 1976.
- [SA90] Stauffer D.,Aharony A., *Intoduction to percolation theory* Taylor&Francis, 1990.
- [Pe02] Peliti L., *Appunti di Maccanica statistica*, 2002.
- [K83] A. Kapitulnik et al J. Phys. A: Math. Gen. 16 L269-L274, 1983.
- [S99] R. Sahara, H. Mizuseki, K. Ohno, and Y. Kawazoe, J. Phys. Soc. Jpn., **68** , 12, 1999.
- [W04] Wanzaller, Cucchieri, Mendes, Krein, Brazilian J. of Phys. **34**, 14 ,2004.
- [H94] Chin-Kun Hu, Chinese J. of Phys. **32**, NO. 5-II, 1994.