

LA PERCOLATION

Année 2013-2014

Par BRUNET Juliette, STERKOS Marie et BARNEOUD Julien élèves de seconde du Lycée d'Altitude de Briançon (Hautes Alpes 05)

Enseignants : Hubert PROAL et Mickaël LISSONDE

Chercheurs : Camille PETIT (Université de Fribourg) et Yves PAPEGAY (INRIA-Sophia Antipolis)

Présentation du sujet

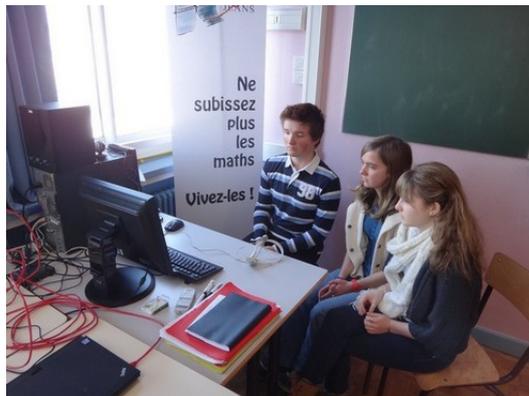
Considérons une grille formée d'un certain nombre de cases noires et blanches. Nous appelons les cases noires les obstacles. On dit qu'il y a **percolation** lorsque l'on peut trouver un chemin formé de cases blanches allant du haut de la grille jusqu'au bas de la grille, ou de manière imagée si un fluide introduit dans la partie haute de la grille trouve un chemin pour atteindre le bas de la grille. Réaliser des expériences de percolation sur des grilles 10x10 en faisant varier la densité d'obstacles (1) dans le but d'établir une loi mathématique qui donnerait la probabilité de percolation en fonction de la densité.

Résultats obtenus

Les simulations réalisées ont permis d'obtenir la forme de la courbe mais ils ne sont pas arrivés à trouver une fonction qui « collerait » le mieux à leurs résultats.

Valorisations des travaux

Les élèves ont présenté leur sujet lors du forum des mathématique d'Aix-en-Provence et lors du congrès *MATh.en.JEANS* 2014 de Lyon.



Groupe en vidéo conférence avec le chercheur



Exposé lors du congrès MeJ de Lyon

I. Sujet

Considérons une grille formée d'un certain nombre de cases noires et blanches. Nous appelons les cases noires les obstacles. On dit qu'il y a **percolation** lorsque l'on peut trouver un chemin formé de cases blanches allant du haut de la grille jusqu'au bas de la grille, ou de manière imagée si un fluide introduit dans la partie haute de la grille trouve un chemin pour atteindre le bas de la grille (voir figures 1 et 2).

Durant cette étude, la probabilité de percolation d'un fluide en fonction de la densité d'obstacles est testée à travers une grille de 10 par 10 puis de 100 par 100 grâce à une programmation informatique.

Les graphiques obtenus ont permis de constater que la probabilité de percolation diminue quand la densité des obstacles augmente. Cette diminution est rapide, visible autour d'un seuil critique.

Cette modélisation nécessite cependant des études complémentaires pour obtenir de meilleurs résultats.

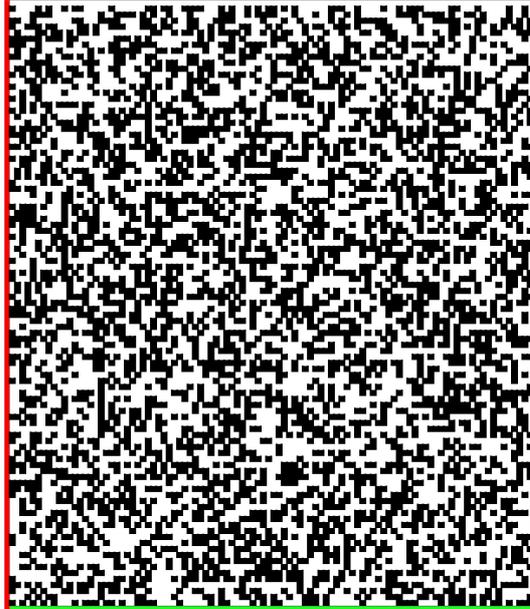


Figure 1 : mélange avec une certaine densité (cases noires). Nous introduisons de l'eau en haut et nous cherchons à savoir si l'eau arrive en bas (zone verte)

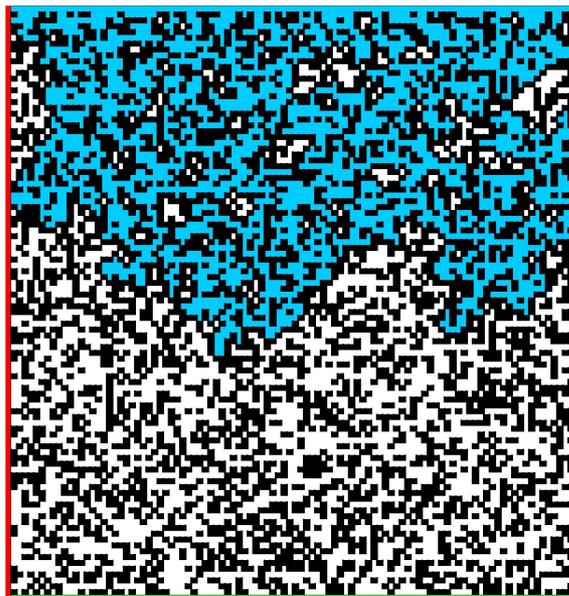


Figure 2 : nous constatons qu'il n'y a pas eu percolation

A travers plusieurs expériences, l'eau peut-elle trouver un chemin entre les obstacles de la grille?

L'eau va-t-elle trouver une issue dans le labyrinthe ?

II. Modélisation (2)

Les premières expériences ont été réalisées avec des grilles de 10 sur 10. Dans une telle grille un certain nombre de cases ont été coloriées en noir au hasard (densité d). Il faut maintenant imaginer qu'il y a un liquide versé en haut de la grille puis regarder si ce liquide peut arriver en bas (figures 3 et 4). Pour chaque expérience, la disposition des obstacles et leur nombre changent.

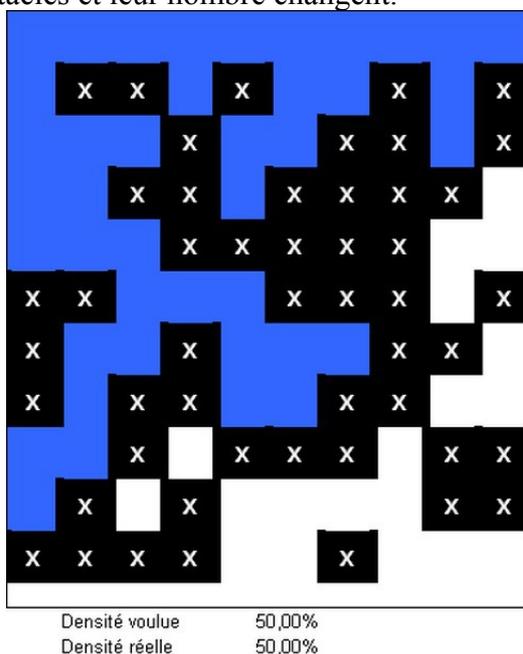


Figure 3 : la densité réelle est de 50% et il n'y a pas percolation

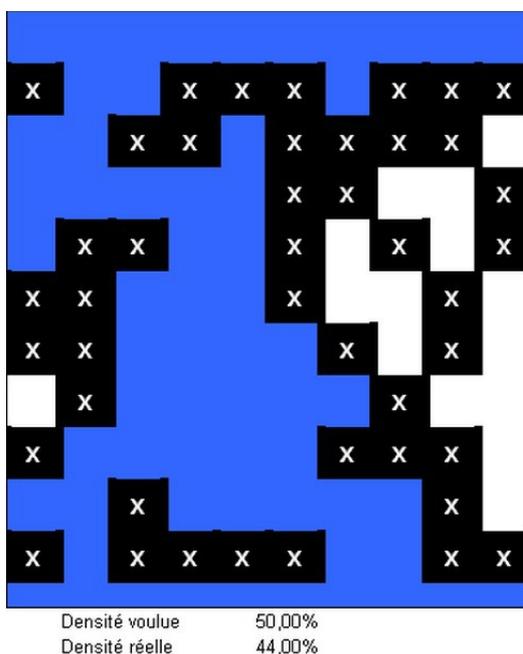


Figure 4 : la densité réelle est de 44% et il y a percolation (le liquide circule du haut vers le bas)

La formule inscrite dans une cellule de la grille est $=SI(ALEA()<=SE\$13;"x";"")$ où la valeur en E13 est la densité voulue.

Étant donné que l'ordinateur met des obstacles au hasard, il est possible que la densité obtenue (c'est-à-dire la densité réelle) soit différente de la valeur fixée au début qui correspond à la densité voulue (figure 5). Pour diminuer cet écart, une solution serait d'augmenter la taille de la grille.

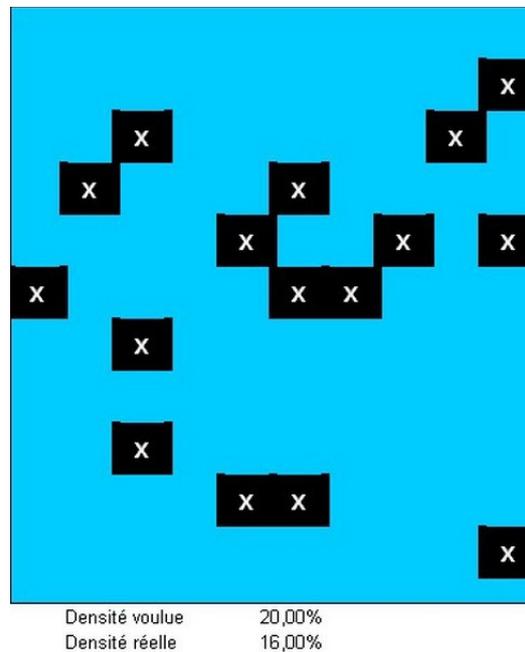


Figure 5 : la densité que nous avons demandée à la machine était de 20%, au final il y a une densité réelle (pourcentage de case occupée) de 16%

III. Résultats expérimentaux

Nos première simulations ont été faites sur tableur, l'ordinateur proposait une grille de 10 sur 10 et nous regardions s'il y avait percolation ou pas.

Pour un grand nombre de densités, nous avons fait à chaque fois 44 expériences et nous avons obtenu notre premier graphique. En abscisse, on a la densité réelle de la grille (c'est-à-dire le nombre d'obstacles) et en ordonnées, la fréquence de percolation qui correspond au nombre de fois où l'eau est passée à travers la grille (figure 6). (3)

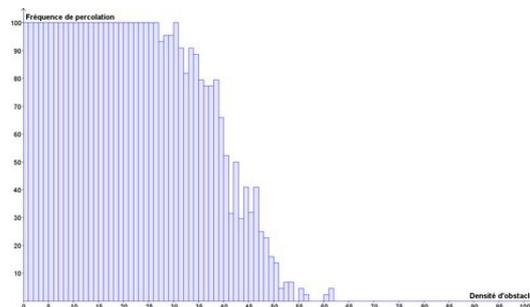


Figure 6 : représentation graphique de nos première simulations sur une grille de 10x10

Un temps considérable a été nécessaire pour obtenir seulement 44 simulations par densité, surtout à cause de la différence entre densité voulue/densité réelle et le recensement des résultats.

Le graphique obtenu est peu précis, avec beaucoup d'irrégularités.

Une programmation en python de ce modèle a permis de réaliser plus d'expériences (voir annexe 1), voici le nouveau graphique (chaque densité a été testée pratiquement 2 000 fois).

Les résultats sont plus précis sur ce graphique en raison du plus grand nombre de simulations réalisées (2 000 pour chaque densité alors qu'il n'y en avait que 44 dans le premier graphique) (figure 7).

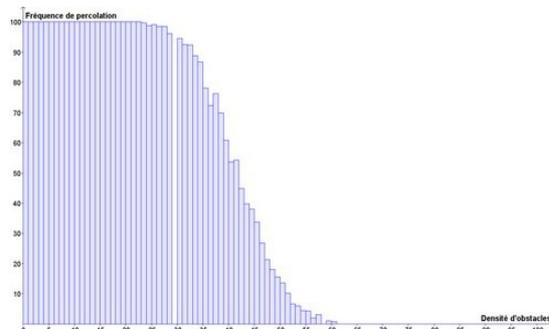


Figure 7 : représentation graphique de nos 2000 simulations sur une grille de 10x10

Si le graphique des 2000 simulations donne une forme plus « lisse », il reste encore quelques irrégularités. Pour des raisons incompréhensibles, les densités 29% et 58% ne semblent pas avoir été testées par le programme (voir annexe 2).

Une nouvelle série avec 4 000 simulations par densité a permis d'obtenir les graphiques suivants (figure 8).

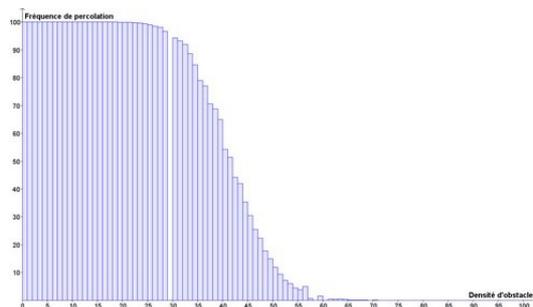


Figure 8 : représentation graphique de nos 4000 simulations sur une grille de 10x10

IV. Analyse des résultats

Au vu de ces résultats, plusieurs observations peuvent être faites :

Propriété 1 : il y aura toujours percolation pour une densité de 0 à 9 sur une grille de 10x10.

En effet, une ligne parfaite constituant un barrage au liquide ne peut pas être formée avec moins de 10 obstacles.

Propriété 2 : dans une grille similaire, il n'y aura jamais percolation pour une densité de 91 à 100.

Dans ce cas, il n'est pas possible alors d'avoir une colonne de 10 cases vides.

Conjecture 3 : la « probabilité de percolation » décroît (« probabilité de non percolation » croît) quand la densité d'obstacles augmente.

Les premières simulations étant insuffisantes, elles ne correspondaient pas exactement à ce constat faute de précision.

Sur les courbes obtenues, la transition entre la phase où la fréquence de percolation est de 100% et la phase où la fréquence de percolation est nulle se fait sur une petite portion de la courbe. De plus les courbes obtenues semblent symétriques par rapport à un point qui se situe dans cette portion. Pour y voir plus clair, des simulations ont été réalisées autour de cette valeur critique qui semble ici être proche de 40 (densité d'obstacles) (figure 9).

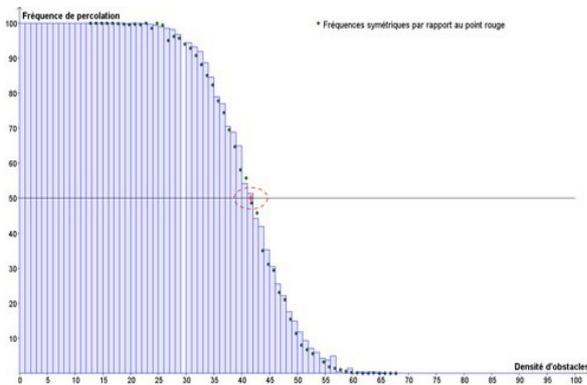
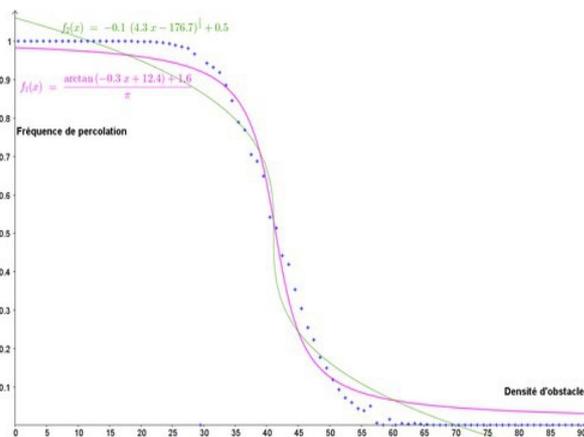


Figure 9 : la courbe semble symétrique par rapport au point rouge (4)

Au premier abord, on pourrait s'attendre à ce que cette valeur critique soit proche de 50 (densité d'obstacles) ce qui correspondrait vraiment au milieu des abscisses (qui vont jusqu'à 100). Le fait que cette valeur soit dans les graphiques précédents proche de 40 peut s'expliquer par les problèmes de bord. En effet, les bords jouent le rôle d'obstacles pour l'eau mais ne sont pas considérés comme étant des obstacles (ils n'influent pas sur la densité d'obstacles de la grille).

Plusieurs fonctions (figure 10) ont été testées pour approcher au mieux les valeurs obtenues. Mais aucune



d'entre elles approche les valeurs.

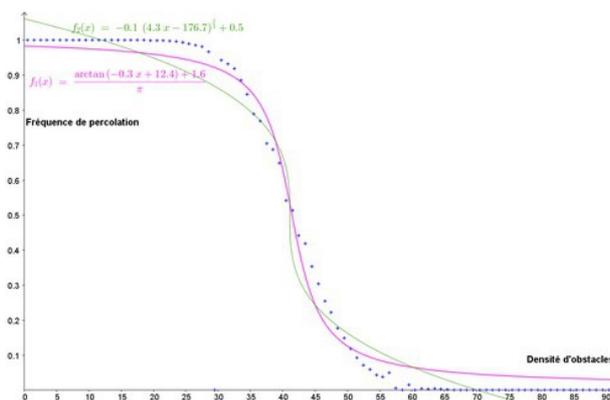


Figure 10 : recherches de fonctions qui « collent » aux valeurs (5)

V. Pistes pour les recherches à venir

Après avoir réalisé ces multiples expériences, les pistes pouvant être envisagées sont encore nombreuses pour trouver de meilleurs résultats :

- Pour éviter les problèmes des bords, simuler avec des grilles plus grandes (100x100).
Le souci est alors que le programme nécessite beaucoup de calculs et le nombre de simulations devra donc être considérablement réduit.
- Imaginer des cellules hexagonales pour avoir un modèle de percolation plus proche de la réalité.
- Ne pas réaliser les expériences sur des grilles carrées mais sur des bandes suffisamment larges pour ne pas avoir des problèmes de bord, par exemple 1000x10 ou encore imaginer que la grille est sur un cylindre.

Récemment, notre enseignant a trouvé un article de la revue [Accromath volume 9 Hiver-Printemps 2014 page 8](#).

Commentaire d'un de nos chercheurs, Camille PETIT : une méthode pour avoir une densité réelle égale à une densité voulue (d) consisterait, par exemple sur une grille 10×10 , à numéroter les cases de 1 à 100 et à en choisir une au hasard, qui va devenir noire. Puis numéroter les cases restantes de 1 à 99 et on en tire une au hasard... Après d étapes, on aura bien une densité exactement égale à d .

Notes d'édition

- (1) La définition de la densité d'obstacles n'est pas donnée ici. On comprend après qu'il s'agit de la proportion de cases noires en pourcentage.
- (2) Les simulations de grilles sont réalisées sur ordinateur. On donne au départ une valeur en pourcentage qui est la proportion de cases noires (obstacles) voulue, appelée densité voulue. L'ordinateur génère un certain nombre de cases noires. La proportion des cases noires générées (en pourcentage) est appelée densité réelle, elle peut être différente de la densité voulue.
- (3) La fréquence de percolation est exprimée en pourcentage.
- (4) Le point rouge a pour coordonnées (41,64;50).
- (5) Il serait intéressant de savoir comment les fonctions f_1 et f_2 du graphique ont été trouvées, car nous ne disposons pas d'explication ici.

Annexe 1

Programme en python pour simuler des percolations

```

from random import random # importer la fonction random pour avoir des valeurs aléatoires

def FMAT(T,p):
# Fabrique MATrice (FMAT) permet de fabriquer une matrice de T sur T avec une densité voulue de p
# La matrice est composée de 1 (obstacles) et 0 (vide)
# Cette fonction va retourner la matrice MAT ainsi que la densité réelle DR
    MAT=[0]*T
    for LI in range(T):
        MAT[LI]=[0]*T
        for CO in range(T):
            q=random()
            if (q<=p):
                MAT[LI][CO]=1
            else:
                MAT[LI][CO]=0
# La matrice MAT est créée, on va calculer sa densité réelle
    S=0
    for LI in range(T):
        for CO in range(T):
            if (MAT[LI][CO]==1):
                S=S+1
    DR=S/(T*T)
    return(MAT,DR)

def FCADRE(MAT):
# Fabrique un CADRE à la matrice MAT avec de l'eau en haut (2), des murs sur le coté (1) et rien en bas (0)
# Cette fonction va retourner la matrice avec son cadre MC de taille (T+2)x(T+2)
    T=len(MAT)
    MC=[0]*(T+2)
    MC[0]=[2]*(T+2)
    for LI in range(T):
        MC[LI+1]=[1]+MAT[LI]+[1]
    MC[T+1]=[0]*(T+2)
    return(MC)

def TMC(MC):
# Test une Matrice Cadrée, c'est-à-dire regarde si l'eau passe du haut vers le bas
# Cette fonction retourne "perco" si elle percolle et "NONp" si elle ne percolle pas
    T=len(MC)-2
    CO1=MC # copie 1 de la matrice initiale
    R="NONp"
    while R=="NONp": # Il va y avoir deux manières d'arrêter la boucle TANT QUE, soit R="perco" si l'eau est arrivée en bas, soit R="STOP" si l'eau n'évolue plus
        CO2=[0]*(T+2) # CO2 va être l'évolution de CO1
        CO2[0]=[2]*(T+2)
        for LI in range(T):
            CO2[LI+1]=[0]*(T+2)
            CO2[LI+1][0]=1
            CO2[LI+1][T+1]=1
            for CO in range(T):
                if (CO1[LI+1][CO+1]==0): # si la case de CO1 est vide (0),
                    if ((CO1[LI+1][CO]==2)or(CO1[LI+1][CO+2]==2)or(CO1[LI][CO+1]==2)or(CO1[LI+2][CO+1]==2)): # il regarde si dans les cases voisines
                        CO2[LI+1][CO+1]=2 # dans ce cas il met de l'eau (2)
                    else:
                        CO2[LI+1][CO+1]=0 # sinon il met la valeur de CO1 (c'est-à-dire 0)
                CO2[LI+1][CO+1]=CO1[LI+1][CO+1] # sinon, si CO1 n'est pas vide (1 ou 2), il met la même valeur à CO2
            for CO in range(T):
                if (CO2[T][CO+1]==2): # Regarde s'il y a de l'eau (2) dans l'avant dernière ligne
                    R="perco" # Dans ce cas il dit qu'il y a percolation et la boucle TANT QUE est finie
                if (R=="NONp"):
                    CO2[T+1]=CO1[T+1]
                    if (CO2==CO1): # Regarde si il y a eu évolution entre les deux étapes, c'est-à-dire si CO1 et CO2 sont identiques
                        R="STOP" # s'il n'y a pas eu d'évolution, il va stopper la boucle TANT QUE
                    else:
                        CO1=CO2 # s'il y a eu évolution CO1 va devenir CO2 et nous allons refaire la boucle
        if (R=="STOP"):
            R="NONp"
    return (R)

# DEBUT DU PROGRAMME

T=int(input ("Taille de la grille ")) # T est la taille de la grille, généralement 10
print()
PO=[0]*(101) # PO[i] va compter le nombre de PERCOLATION pour une densité de i
PN=[0]*(101) # PN[i] va compter le nombre de NON PERCOLATION pour une densité de i
for j in range(101): # j va passer toutes les densités de 0 à 100
    p=j/100
    for i in range(2000): # fait 2000 simulations de densité voulue j (attention ce n'est pas forcément la densité réelle)
        M=FMAT(T,p) # M[0] va être une grille aléatoire de T sur T et M[1] sa densité réelle
        D=int(M[1]*100)
        MC=FCADRE(M[0]) # fabrication d'un cadre autour de M[1], puis test le cadre
        if (MC=="NONp"):
            PN[D]=PN[D]+1
        else:
            PO[D]=PO[D]+1
print("RESULTATS")
for i in range(101):
    print ("Densité -",i," - Non percolation-",PN[i]," - percolation-",PO[i]) # affiche les résultats, compter 3 minutes pour des grilles de 10x10

```

Annexe 2
Exemple de résultats

Densité - 17	- Non percolation-	0	- percolation-	2017
Densité - 18	- Non percolation-	1	- percolation-	1932
Densité - 19	- Non percolation-	2	- percolation-	1912
Densité - 20	- Non percolation-	3	- percolation-	2034
Densité - 21	- Non percolation-	1	- percolation-	2049
Densité - 22	- Non percolation-	4	- percolation-	1934
Densité - 23	- Non percolation-	5	- percolation-	1981
Densité - 24	- Non percolation-	14	- percolation-	1978
Densité - 25	- Non percolation-	21	- percolation-	1959
Densité - 26	- Non percolation-	30	- percolation-	1952
Densité - 27	- Non percolation-	42	- percolation-	1901
Densité - 28	- Non percolation-	130	- percolation-	3855
Densité - 29	- Non percolation-	0	- percolation-	0
Densité - 30	- Non percolation-	104	- percolation-	1928
Densité - 31	- Non percolation-	122	- percolation-	1854
Densité - 32	- Non percolation-	174	- percolation-	1829
Densité - 33	- Non percolation-	242	- percolation-	1651
Densité - 34	- Non percolation-	315	- percolation-	1664
Densité - 35	- Non percolation-	388	- percolation-	1593
Densité - 36	- Non percolation-	476	- percolation-	1499
Densité - 37	- Non percolation-	539	- percolation-	1423
Densité - 38	- Non percolation-	655	- percolation-	1291
Densité - 39	- Non percolation-	766	- percolation-	1169
Densité - 40	- Non percolation-	880	- percolation-	1157
Densité - 41	- Non percolation-	969	- percolation-	975
Densité - 42	- Non percolation-	1074	- percolation-	901
Densité - 43	- Non percolation-	1213	- percolation-	843
Densité - 44	- Non percolation-	1264	- percolation-	661
Densité - 45	- Non percolation-	1342	- percolation-	610
Densité - 46	- Non percolation-	1471	- percolation-	542
Densité - 47	- Non percolation-	1549	- percolation-	449
Densité - 48	- Non percolation-	1590	- percolation-	362
Densité - 49	- Non percolation-	1666	- percolation-	333
Densité - 50	- Non percolation-	1763	- percolation-	271
Densité - 51	- Non percolation-	1820	- percolation-	182
Densité - 52	- Non percolation-	1806	- percolation-	179
Densité - 53	- Non percolation-	1918	- percolation-	127
Densité - 54	- Non percolation-	1897	- percolation-	88
Densité - 55	- Non percolation-	1948	- percolation-	67
Densité - 56	- Non percolation-	3817	- percolation-	83
Densité - 57	- Non percolation-	1924	- percolation-	29
Densité - 58	- Non percolation-	0	- percolation-	0
Densité - 59	- Non percolation-	1975	- percolation-	21
Densité - 60	- Non percolation-	1979	- percolation-	12
Densité - 61	- Non percolation-	1916	- percolation-	9
Densité - 62	- Non percolation-	1947	- percolation-	0
Densité - 63	- Non percolation-	1980	- percolation-	10