

# Моделирование процесса перколяции

Мейер Никита Владимирович

Санкт-Петербургский государственный университет  
Прикладная математика и информатика  
Вычислительная стохастика и статистические модели

Научный руководитель: д. ф.-м. н., профессор Ермаков С. М.  
Рецензент: к. ф.-м. н., Дмитриев А. В.



2018 г.

# Общая постановка задачи теории перколяции

- Перколяционная решетка является математической моделью множества физических процессов (просачивание через пористый материал, электропроводность в полупроводниках и другие);
- Решетка  $n \times n$  (для простоты квадратная),  $n$  — размерность решетки;
- Узлы решетки заполняются независимо с вероятностью  $p$ ;
- Заполненные узлы принадлежат одному кластеру, если их можно соединить цепочкой соседних заполненных узлов.

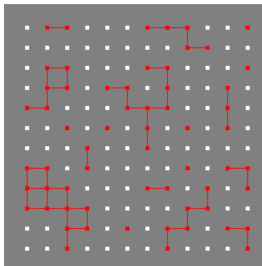


Рис.: Пример решетки. Красным отмечены заполненные узлы. Соединены соседние заполненные узлы, принадлежащие одному кластеру.

- Задача:
  - получение значения среднего числа кластеров на решетке с помощью моделирования;
- Применение:
  - выведение на орбиту большого количества наноспутников;
  - обеспечение безопасности крупных компьютерных сетей;
- Задача может быть решена стандартными алгоритмами теории перколяции;
- Из-за большого вычислительного объема встает вопрос о разработке алгоритма, позволяющего получать результат с необходимой точностью на выборке меньшего объема.

Рассмотрим заполнение узлов на решетке как случайный эксперимент. Определим вероятностное пространство данного эксперимента:

- $\Omega = \{\omega : \omega = (c_1, \dots, c_{n^2}), c_i = \{0; 1\}\}$ ;
- $|\Omega| = 2^{n^2}$  — количество различных элементарных исходов;
- $\mathfrak{A} = \{A_i \subseteq \Omega\}$ ;
- $\mathbb{P}(\{\omega\}) = p^m(1-p)^{n^2-m}$ , где  $m$  — количество  $c_i$ , равных 1 (т.е. количество заполненных узлов элементарного исхода);
- Дискретная случайная величина  $K(\omega) : \Omega \rightarrow \mathbb{N}$ , численно равная количеству кластеров на решетке.

- Вместо вероятностного пространства  $\{\Omega, \mathcal{A}, \mathbb{P}\}$  рассматривается  $\{\Omega, \mathcal{A}, \mathbb{Q}\}$ ;
- Вероятности элементарных исходов меняются с  $p_i = \mathbb{P}(\omega_i)$  на  $q_i = \mathbb{Q}(\omega_i)$ ;

## Определение

*Весом элементарного исхода  $\omega_i$  при замене меры с  $\mathbb{P}$  на  $\mathbb{Q}$  называется  $W_i = \frac{\mathbb{P}(\omega_i)}{\mathbb{Q}(\omega_i)} = \frac{p_i}{q_i}$ ;*

- Введем на  $\{\Omega, \mathcal{A}, \mathbb{Q}\}$  дискретную случайную величину  $K'(\omega_i) : \Omega \rightarrow \mathbb{N}$ , определяемую как  $K'(\omega_i) = K(\omega_i)W_i$ ;
- Можно показать, что  $\mathbb{E}(K') = \mathbb{E}(K)$ .

Воспользуемся методом Монте-Карло:

- Введем оценку:

$$\hat{K}' = \frac{1}{N} \sum_{i=1}^N K'_i,$$

где  $K'_1, \dots, K'_N$  — независимые случайные величины, распределенные как  $K'$ ;  $N$  — их количество;

- $\mathbb{E}(\hat{K}') = \mathbb{E}(K') = \mathbb{E}(K)$ ;
- Доверительный интервал уровня  $1 - \gamma$  при больших  $N$ :

$$P \left( \hat{K}' - x_\gamma \frac{\sigma}{\sqrt{N}} < \mathbb{E}(K') < \hat{K}' + x_\gamma \frac{\sigma}{\sqrt{N}} \right) \approx 1 - \gamma,$$

где  $x_\gamma$  — решение уравнения  $2\Phi(x) = 2 - \gamma$ ;  $\Phi(x)$  — функция распределения  $N(0, 1)$ ;  $\sigma = \sqrt{\text{Var}(K')}$ .

Для моделирования воспользуемся алгоритмом Хошена-Копельмана (J. Hoshen and R. Kopelman, 1976):

- **Входные данные:** размерность решетки  $n$ ; вероятность заполненного узла  $p$ .
- **Тело алгоритма:**
  - Элементы массива, каждый из которых соответствует узлу решетки, заполняются поочередно в заданном порядке (реализация равномерного распределения на  $[0, 1]$  сравнивается с  $p$ ).
  - Для каждого заполненного узла определяется его кластерная метка (номер кластера, которому он принадлежит) через кластерные метки соседних узлов.
- **Результат работы алгоритма:** количество различных кластерных меток на решетке.

# Программная реализация алгоритма Хошена-Копельмана

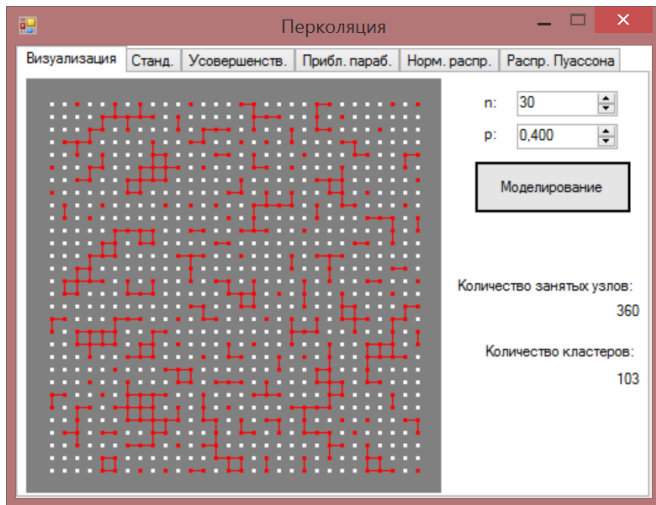


Рис.: Визуализация заполненной перколяционной решетки.



- **Входные данные:** вероятность заполненного узла  $p$ ; замещающая вероятность  $q$ ; размерность решетки  $n$ .
- **Тело алгоритма:**
  - Применяется алгоритм Хошена-Копельмана, входными данными для которого будет  $q$  и  $n$ . Количество кластерных меток обозначается за  $d$ .
  - Рассчитывается вес элементарного исхода

$$W = \frac{p^m (1 - p)^{n^2 - m}}{q^m (1 - q)^{n^2 - m}},$$

количество заполненных узлов  $m$  определяется по количеству ненулевых элементов массива кластерных меток.

- **Результат работы алгоритма:** величина  $b = Wd$ .

# Программная реализация усовершенствованного алгоритма Хошена-Копельмана

Перколяция

Визуализация | Станд. | Усовершенств. | Прибл. параб. | Норм. распр. | Распр. Пуассона

Размерность решетки,  $n$ : 20

Вероятность заполнения узла,  $p$ : 0.10000

Замещающая вероятность,  $q$ : 0.10170

Объем выборки,  $N$ : 50000

☒ Выборочная дисперсия

☒ Моделировать стандартным алгоритмом Хошена-Копельмана

Моделирование

Усовершенствованный алгоритм:	Стандартный алгоритм:
Выборочное среднее: 32,4387923538898	Выборочное среднее: 32,435
Выборочная дисперсия: 6,18943511270417	Выборочная дисперсия: 19,9074931498523
Выборочное отклонение: 2,48785753464787	Выборочное отклонение: 4,46178138750121

Рис.: Реализация оценки и вычисление несмещенной выборочной дисперсии усовершенствованным и стандартным алгоритмом.

# Поиск оптимальной замещающей вероятности

- Моделирование решетки ( $n = 20$ ;  $p = 0.1$ ;  $N = 100000$ ) для различных замещающих вероятностей  $q$ ;
- Задача: поиск оптимальной замещающей вероятности  $q_0$ , для которой достигается минимум несмещенной выборочной дисперсии  $S^2$ .

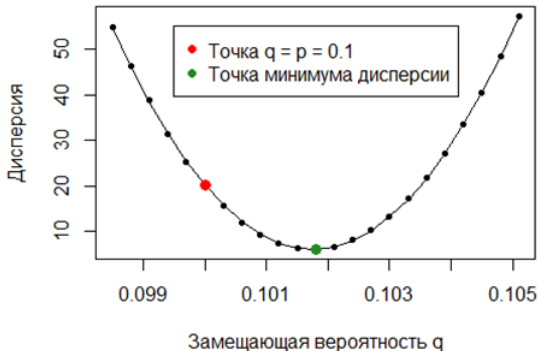


Рис.: Несмещенная выборочная дисперсия  $S^2$  для различных замещающих вероятностей  $q$ . Полученные значения приблизительно описываются параболой.

- **Входные данные:** размерность решетки  $n$ ; вероятность заполненного узла  $p$ ; замещающие вероятности  $q_1, q_2, q_3$ ; объем выборки  $N$ .
- **Тело алгоритма:**
  - 1 По каждой из замещающих вероятностей  $q_1, q_2, q_3$  с помощью усовершенствованного алгоритма Хошена-Копельмана реализуется выборка объема  $N$ . Для каждой выборки вычисляется несмещенная выборочная дисперсия  $S_1^2, S_2^2, S_3^2$ .
  - 2 Строится квадратичная функция  $f(x) = ax^2 + bx + c$ , решениями которой являются  $\{(q_1, S_1^2); (q_2, S_2^2); (q_3, S_3^2)\}$ .
  - 3 Вычисляются координаты вершины параболы  $(q_0, f(q_0))$ .
  - 4 По замещающей вероятности  $q_0$  с помощью усовершенствованного алгоритма Хошена-Копельмана реализуется выборка объема  $N$ . Для этой выборки вычисляется несмещенная выборочная дисперсия  $S_0^2$ .
- **Результат работы алгоритма:**  $q_0, f(q_0), S_0^2$ .

# Программная реализация метода приближения параболой

Перколяция

Визуализация   Станд.   Усовершенств.   Прибл. параб.   Норм. распр.   Распр. Пуассона

Размерность решетки, n: 20

Вероятность заполнения узла, p: 0,10000

Объем выборки, N: 50000

Замещающая вероятность, q2: 0,10300

Замещающая вероятность, q3: 0,10600

Моделирование

Выборочная дисперсия для вероятности p: 20,0833416668333

Выборочная дисперсия для замещающей вероятности q2: 13,3598187103751

Выборочная дисперсия для замещающей вероятности q3: 88,347753470931

Оптимальная замещающая вероятность, q0: 0,101746851169138

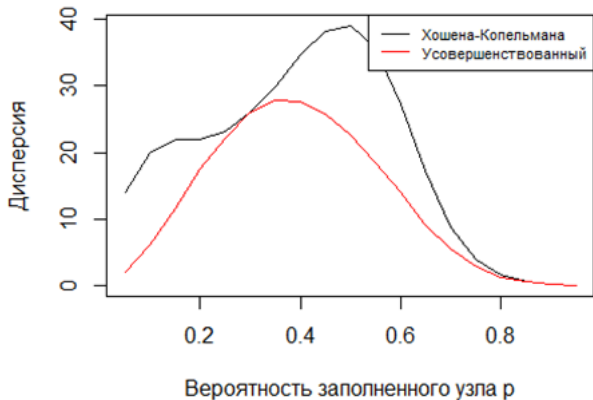
Значение параболы в точке q0: 6,23102972352353

Выборочная дисперсия для замещающей вероятности q0: 6,04644102165684

Рис.: Вычисление координат вершины параболы и сравнение значения, полученного с помощью моделирования, с теоретическим.

# Сравнение с алгоритмом Хошена-Копельмана

Сравним значения  $S^2$  обычного и усовершенствованного алгоритма с замещающей вероятностью  $q_0$  на решетке ( $n = 20$ ;  $N = 50000$ ) для различных  $p$ .



**Рис.:** Усовершенствованный алгоритм позволяет получать оценки с существенно меньшей дисперсией.

Алгоритм Хошена-Копельмана позволяет задавать вероятностные меры вида  $\mathbb{P}(\{\omega\}) = p^m(1-p)^{n^2-m}$ , где  $m$  — количество заполненных узлов элементарного исхода.

Сформулируем алгоритм, с помощью которого можно задавать более широкий класс случайных величин  $K'$ .

- **Входные данные:** размерность решетки  $n$ ; распределение случайной величины  $R$ , принимающей целые значения от 0 до  $n^2$ .
- **Тело алгоритма:**
  - 1 По заданному распределению получается реализация  $m$  случайной величины  $R$ .
  - 2 Моделирование решетки, на которой случайным образом заполнено ровно  $m$  узлов.
- **Результат работы алгоритма:** количество кластерных меток.

# Моделирование с помощью алгоритма, расширяющего класс мер

- Вероятность элементарного исхода  $\omega$ :

$$\mathbb{Q}(\omega) = \frac{P(R = m)}{C_{n^2}^m} > 0,$$

где  $m$  — число заполненных узлов в элементарном исходе  $\omega$ ;

- Вес  $W$  элементарного исхода  $\omega$ :

$$W = \frac{p^m (1 - p)^{n^2 - m} C_{n^2}^m}{P(R = m)};$$

- Моделируем с помощью метода существенной выборки для различных распределений  $R$ .



# Дискретный аналог нормального распределения

Возьмем в качестве  $R$  следующее распределение с задаваемыми параметрами  $a$  и  $\sigma$ :

$$P(R = m) = \frac{e^{-\frac{(m-np-a)^2}{2\sigma^2}}}{\sum_{i=0}^{n^2} e^{-\frac{(i-np-a)^2}{2\sigma^2}}}$$



Рис.: Значения несмещенной выборочной дисперсии  $S^2$  для различных  $p$ .

# Усеченное распределение Пуассона

Возьмем в качестве  $R$  следующее распределение с задаваемым параметром  $\lambda$ :

$$P(R = m) = \frac{\frac{\lambda^m}{m!}}{\sum_{i=0}^{n^2} \frac{\lambda^i}{i!}}$$

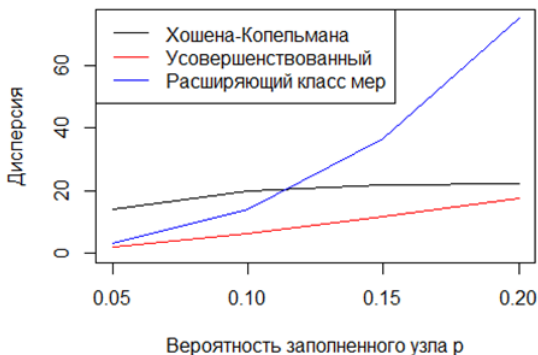


Рис.: Значения несмещенной выборочной дисперсии  $S^2$  для различных  $p$ .

В моей работе:

- программно реализован алгоритм Хошена-Копельмана;
- разработан и реализован усовершенствованный алгоритм моделирования среднего числа кластеров;
- разработан и реализован метод приближения параболой для поиска оптимальной замещающей вероятности;
- найдены оптимальные вероятности, позволяющие реализовывать оценку с меньшей дисперсией;
- разработан и реализован алгоритм, расширяющий класс мер;
- подобраны два распределения, позволяющие реализовывать оценку с меньшей дисперсией, используя алгоритм, расширяющий класс мер;
- разработан подход к уменьшению дисперсии в подобных задачах, который может быть использован в будущем для дальнейшей оптимизации имеющихся алгоритмов.