Теория по проекту
«Псевдослучайные структуры точек на поверхности»

Авторы: Дьяков Илья, Иванов Егор

Осень 2023

1 Цель работы

Цель работы – показать, насколько искажено (может, и нет) восприятие человеком случайности путем создания компьютерной симуляции-игры со следующим сценарием: пользователю предлагается изучить набор точек, распределенных по плоскости, и сказать, случаен он или нет.

В данном случае под случайностью набора следует понимать полное отсутствие каких-либо ограничений (априорных или изменяющихся со временем) на координаты появляющихся точек. Иными словами, каждая из координат всех изображенных точек – это реализация равномерно распределенной случайной величины. В прочих страгетиях генерации случайность сохраняется, но появляются визуально отличимые (может, и не очень) особенности. Следует пояснить, что накладываемые в таких стратегиях ограничения не слишком сложны и могут быть даже в точности установленны наблюдателем.

Здесь авторам кажется уместной следующая фантазия или сюжет игры, если угодно. Пользователь может представить себя главой научной группы в крупном исследовательском центре. У него «на руках» результаты некоторого эксперимента (сгенерированные симуляцией точки), по которым нужно решить, есть ли в данных некоторая зависимость, которую стоит дополнительно исследовать (что значит тратить на это, например, человеко-часы), или таковой нет и следует искать что-то другое.

Кроме того, целью создания симуляции является желание показать, насколько интересными могут быть картины, составленные из «обычных» случайных (или не очень, это уже вопрос к игроку) точек. Оказалось, что использование даже не очень сложных стратегий может порождать удивительно красивые паттерны. В целом случайные точки для человеческого взгляда кажутся не только приятными, но и даже аппетитными и загадочными (см. Рис. 1, 2).

2 О программе

В данном разделе описаны возможности взаимодействия пользователя с игрой. Так как уважаемый читатель уже получил доступ к данному документу, считаем, что с главным меню он освоился. Чтобы перейти к симуляции, следует нажать на кнопку «Модель». После этого должно появиться окно, в правой части которого расположено «игровое поле», а в левой – панель управления (см. Рис. 3).

а)

б)

Рис. 1: «Вкусные» случайные точки

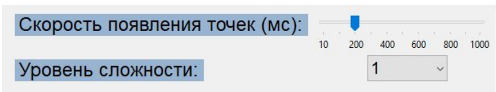
2.1 Кнопки

- **«Скорость появления точек (мс)»** – слайдер для величины в миллисекундах интервала времени между появлением следующей и предыдущей точек
- **«Уровень сложности»** – изменение уровня сложности (см. подробнее в секции 4 «Документация»)
- **«Сгенерировать»** – после нажатия начинается (или продолжается) генерация точек
- **«Стоп»** – при нажатии генерация останавливается (и может быть продолжена с помощью нажатия на кнопку **«Сгенерировать»**)
- **«Новый метод»** – все уже сгенерированные точки стираются, случайно выбирается новая стратегия генерации (может быть запущена нажатием кнопки **«Сгенерировать»**)
- **«Очистить»** – после нажатия стираются все сгенерированные ранее точки (текущий процесс генерации не останавливается)
- **«Выбрать метод вручную»** – после нажатия появляется новое окно с «ручным» выбором метода генерации. НЕ РЕКОМЕНДУЕТСЯ нажимать, если пользователь еще «не поиграл» (или не поугадывал)
- **«Показать текущий метод»** – после нажатия появляется окно с отгадкой (с, возможно, не слишком крупным шрифтом, однако это зависит от настроек всех всплывающих окон операционной системы, которые в случае необходимости могут быть изменены пользователем), а на поле появляются элементы, поясняющие стратегию

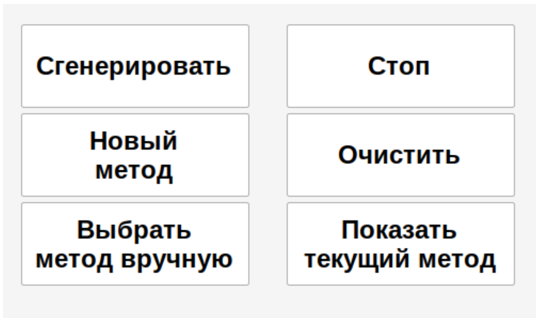
3 О случайности

Здесь пользователю представлены рассуждения о случайности, наличие или отсутствие которой ему и предлагается обнаружить. Основная идея: показать, что стоит за случайностью, предоставляемой компьютером, как с ней можно работать и можно ли ее вообще считать случайностью.

Рис. 2: «Загадочные» случайные точки



а)



б)

Рис. 3: Панель управления

3.1 Модель и реальность

Вспомним основной математический объект, связанный с описанием случайности, вероятностную тройку:

$$(\Omega, \mathcal{F}, \mathsf{P})$$

Здесь

Ω – множество элементарных исходов

$\mathcal{F}$ – сигма-алгебра (или алгебра) подмножеств множества элементарных исходов

P – вероятностная мера

Несложно объяснить, почему модель именно такая. Нужно описывать случайные эксперименты, для которых естественны воспроизводимость (в смысле возможности многократного повторения) и недетерминированность (невозможность предсказать заранее результат, например, в силу большой вычислительной сложности этой операции).

Эксперимент – очень общее понятие, поэтому Ω не имеет никакой наперед заданной структуры и состоит из возможных реализаций ω эксперимента. Теперь мы, как исследователи, хотели бы получать ответ на вопрос вида: «Имеет ли данная реализация некоторое свойство?» или «Какова вероятность того, что данная реализация будет обладать некоторым свойством?» То есть нас интересует не конкретный элементарный исход (вероятность которого в, например, непрерывных моделях вовсе равна 0), а его попадание во множество элементарных исходов, объединенных по некоторому признаку. Именно поэтому область определения вероятности это набор подмножеств $\mathcal{F}$. Руководствуясь такой логикой, несложно объяснить необходимость всех свойств, требуемых в определениях сигма-алгебры (алгебры) и вероятностной меры.

Следующий шаг формализации – понятие случайной величины, которое идейно нужно для «побега» из неизвестной структуры Ω и $\mathcal{F}$ в хорошо изученные пространства, например, $\mathbb{R}^n$. Мы забываем про эксперимент, наш интересует лишь распределение случайной величины, в нем появляющейся. Этот переход позволяет строить математическую теорию и доказывать теоремы, однако на практике никуда «сбежать» не получается, потому что в реальном мире существует именно конкретный исход ω и возможность его многократной генерации: $\omega_1, \omega_2, \dots, \omega_N, N \gg 1$.

Хороший пример для пояснения – это игральные кости, комбинаторные задачи с которыми решаются мелом на доске, а проверку полученных результатов можно провести, подкидывая настоящий кубик и фиксирируя результаты на бумаге. При должном стремлении можно проверить справедливость, например, закона больших чисел или центральной предельной теоремы. Но что делать, если например, был получен теоретический результат о поведении случайных величин не с легко реализуемым распределением. Как провести эксперимент? Как сгенерировать реализацию случайной величины, имеющей, например, нормальное или гамма распределение?

3.2 О генерации случайных величин

Рассмотрим следующий теоретический результат:

Теорема 3.1 (Преобразование Бокса — Мюллера)**.** Пусть r и φ – независимые случайные величины, равномерно распределенные на полуинтервале $(0, 1]$. Тогда случайные величины

$$z_0 = \cos(2\pi\varphi)\sqrt{-2\ln r}$$
$$z_1 = \sin(2\pi\varphi)\sqrt{-2\ln r}$$

независимы и имеют стандартное нормальное распределение.

Идея: генерировать на компьютере реализации равномерно распределенных случайных величин, а потом, используя теоретические результаты, получать необходимые распределения. Таким образом, достаточно научиться моделировать равномерное распределение, но как уже было упомянуто, в реальном мире это можно сделать только с помощью проведения экспериментов. Здесь существуют два подхода:

3.2.1 «Честный» подход

Пользоваться виртуальным устройством `/dev/random`, по сути являющимся бинарным файлом, который заполняется результатами побочных вычислений прочих системных программ. Это *настоящая* случайность, которую можно считать реализацией достаточно мелкой дискретизации равномерного распределения.

3.2.2 «Нечестный» подход

Идея проста: генерировать не то, что нужно, а что-то, что похоже на то, что нужно, причем делать это по алгоритму. Опишем один из них – так называемый линейный конгруэнтный метод.

Выбираются 4 числа:

1. $m > 0$ – модуль
2. $0 \leq a < m$ – множитель
3. $0 \leq c < m$ – приращение
4. X_0 – начальное значение

После чего очередное «случайное» число вычисляется по следующему правилу:

$$X_{n+1} = (aX_n + c) \mod m$$

Стоит отметить, что данный подход имеет свои минусы. Так, им не стоит пользоваться в криптографии в силу малой надежности (достаточно знать параметры, часто используемые наборы которых известны). Однако, он прост в реализации и обеспечивает воспроизводимость экспериментов.

4 Документация

ВАЖНО! Далее будут описаны конкретные методы построения структур точек, поэтому если вы еще не «поиграли», советуем в этот раздел НЕ ЗАГЛЯДЫВАТЬ!

4.1 Общие положения

У каждой из стратегий симуляции есть 3 уровня сложности для угадывания: простой (1), средний (2) и сложный (3). Сопроводим их словесным описанием:

- простой (1): легко угадываются и отсутствие случайности, и сама стратегия
- средний (2): несложно устаналивается отсутствие случайности, стратегию найти сложно, но по-прежнему возможно
- сложный (3): сложно определяется неслучайность, стратегию определить практически невозможно

Ниже приведены словесные описания стратегий, а также примеры работы симуляции при их генерации. На каждом из рисунков 1-6 изображения а), б) и в) – примеры работы на простом, среднем и сложном уровнях соответственно. Если есть изображение г), то это графическое пояснение к стратегии.

4.2 Стратегии

4.2.1 Правильные шестиугольники

Точки расположены в вершинах равносторонних шестиугольников, замощающих поверхность. Шестиугольная сетка выводится на поле после нажатия на кнопку **«Показать текущий метод»**.

4.2.2 Правильные треугольники

Точки расположены в вершинах равносторонних треугольников, замощающих поверхность. Треугольная сетка выводится на поле после нажатия на кнопку **«Показать текущий метод»**.

4.2.3 Отрицательная корреляция (оттакливание)

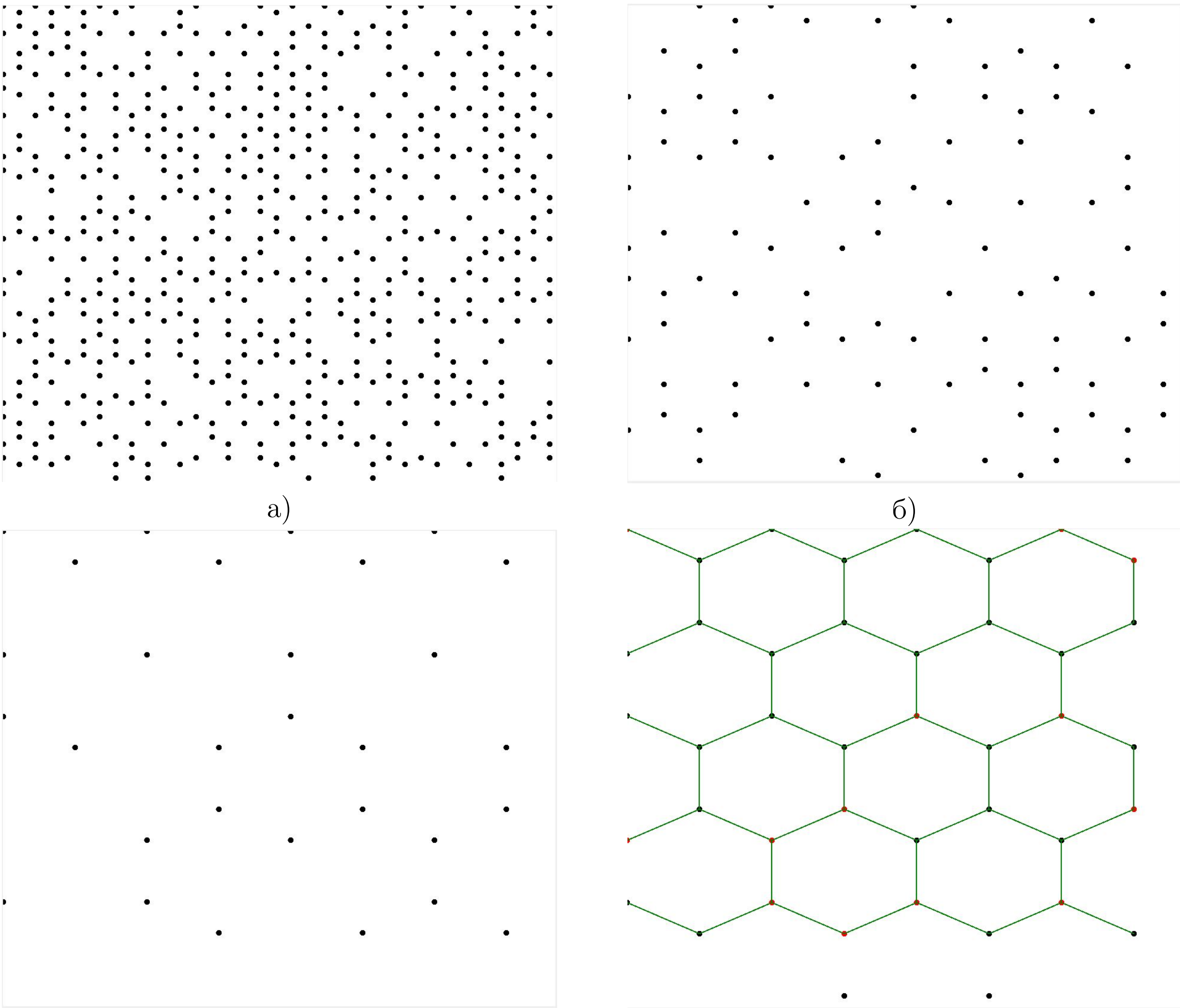
Вокруг каждой новой сгенерированной точки появляется (незримо для пользователя) окрестность фиксированного радиуса, в которую не могут попасть следующие точки. Границы окрестностей выводятся на поле при нажатии на кнопку **«Показать текущий метод»**.

4.2.4 Положительная корреляция (притяжение)

Вновь сгенерированная точка увеличивает вероятность появления следующих вокруг себя. Реализация – дискретная, т.е. поле разбивается на множество достаточно малых подпрямоугольников, для которых и меняются вероятности.

4.2.5 Фракталы (треугольники Серпинского)

Область на экране разделена на три постепенно появляющихся треугольника Серпинского. Оказывается, что этот самоподобный объект несложно построить с помощью аналога случайного блуждания. Для этого нужно фиксировать положения точки, движущейся по следующем правилу: на каждом шаге равновероятно выбирается одна из трех опорных точек, после чего точка приближается к ней на половину от расстояния до нее.



а) б) в) г)

Рис. 4: Стратегия «Правильные шестиугольники»

4.2.6 Случайное блуждание

Точки на поверхности – это траектория случайно блуждающей точки: на каждом шаге равновероятно выбирается угол смещения, модуль перемещения остается постоянным.

4.2.7 Концентрические окружности

Точки появляются только на нескольких окружностях с общим центром в середине игрового поля. Окружности выводятся на поле нажатия на кнопку **«Показать текущий метод»**.

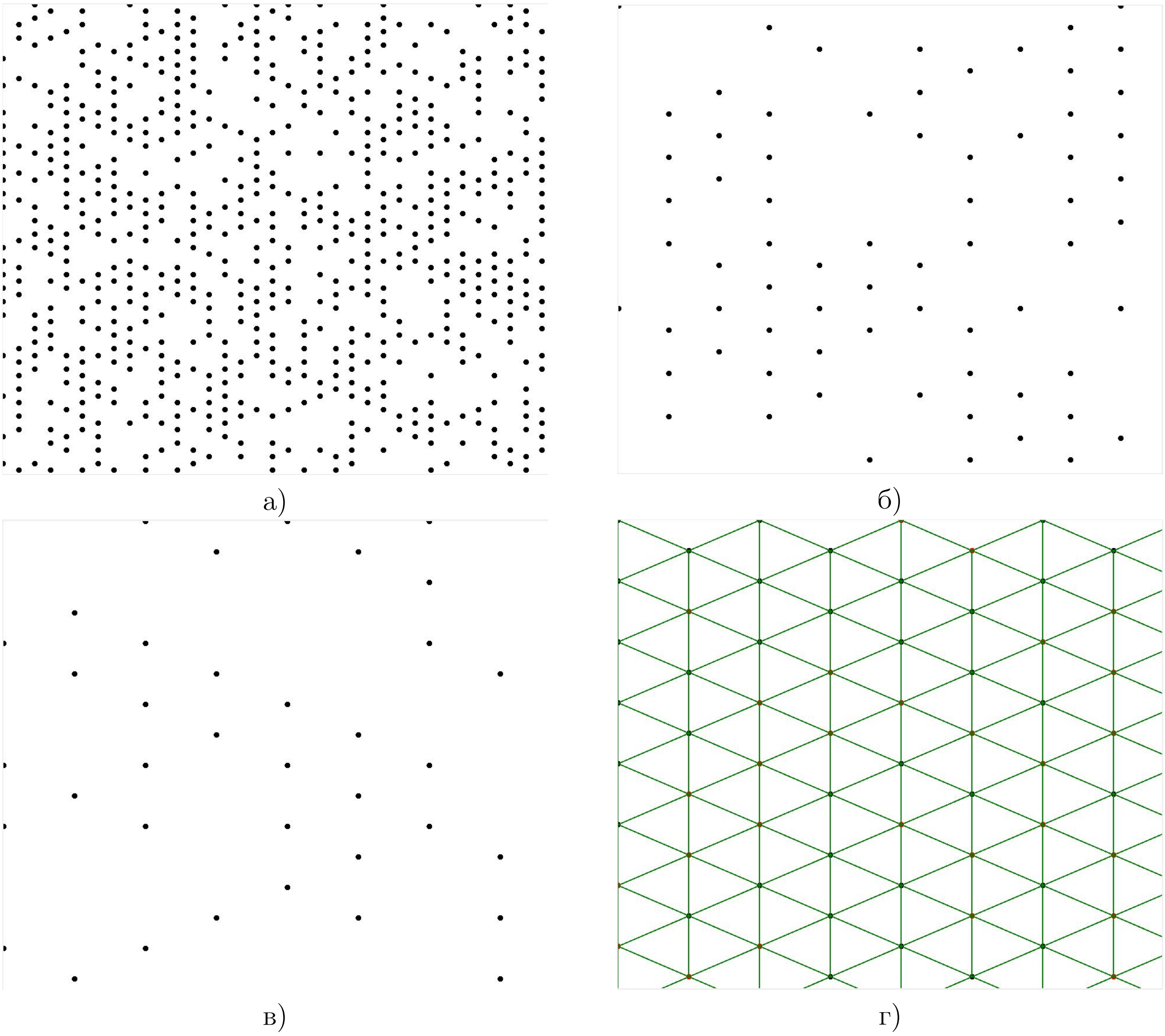
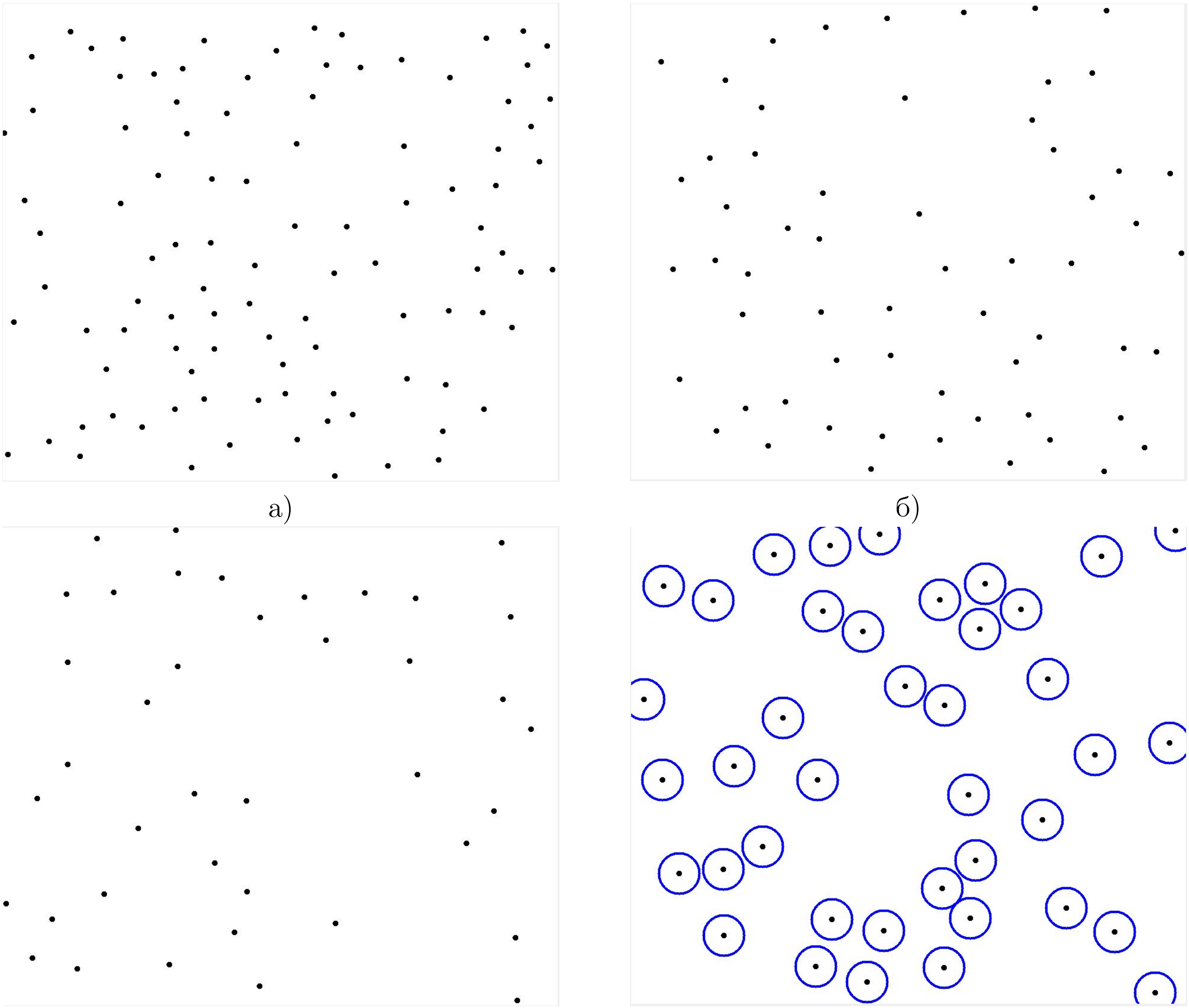


Рис. 5: Стратегия «Правильные треугольники»

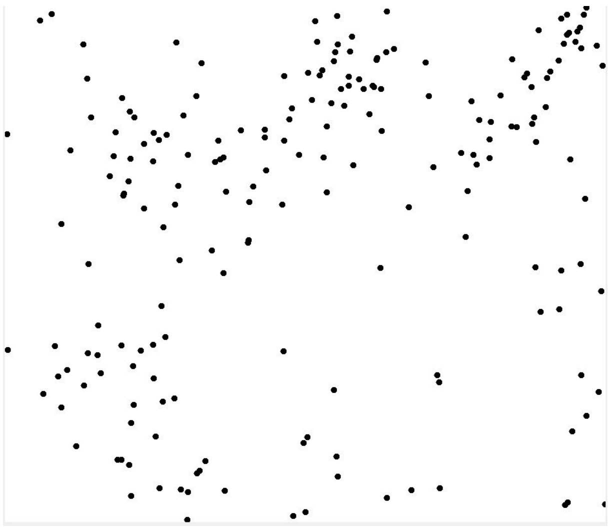


а) б) в) г)

Рис. 6: Стратегия «Отрицательная корреляция»

а)

б)



в)

Рис. 7: Стратегия «Положительная корреляция»

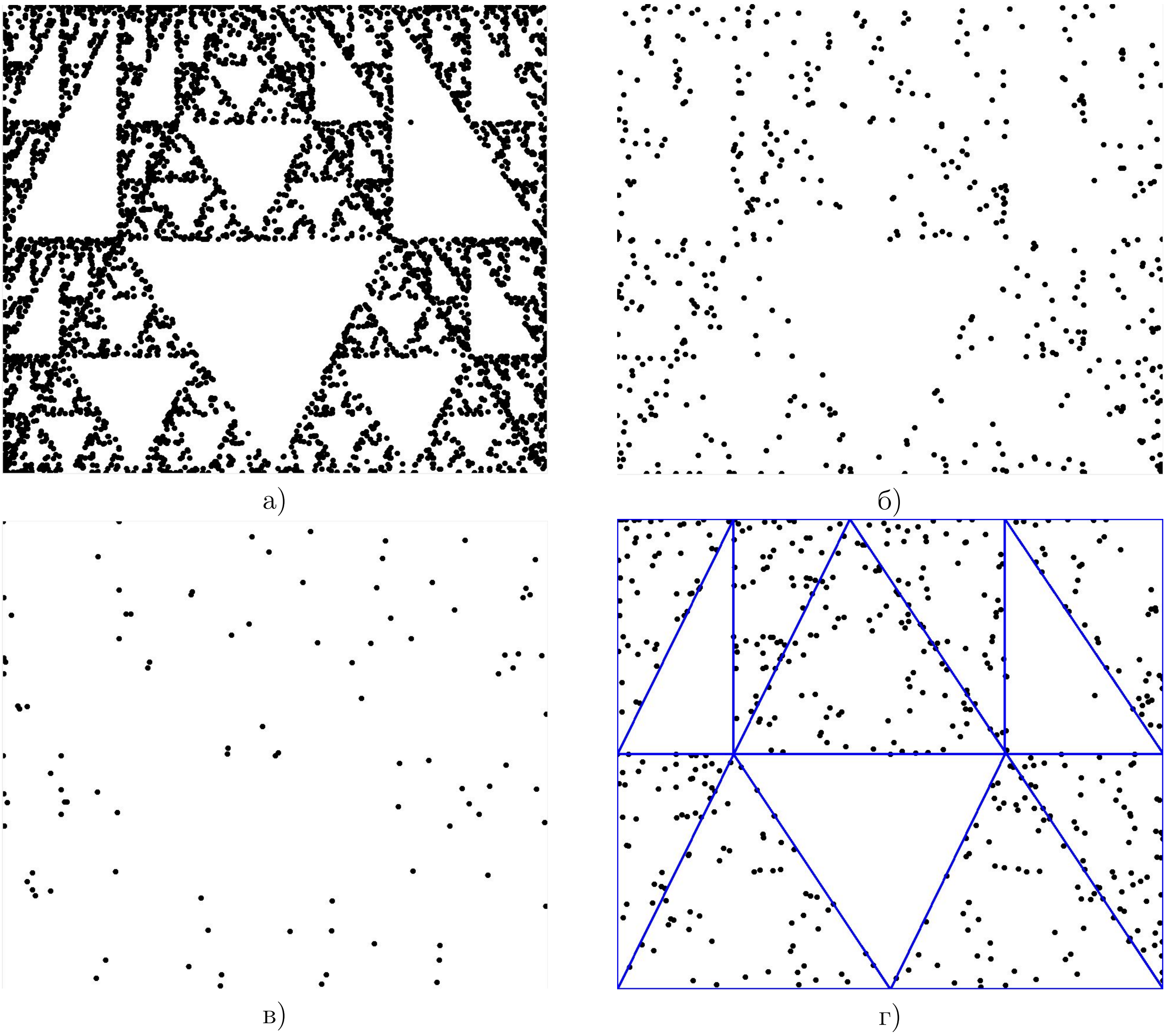


Рис. 8: Стратегия «Треугольники Серпинского»

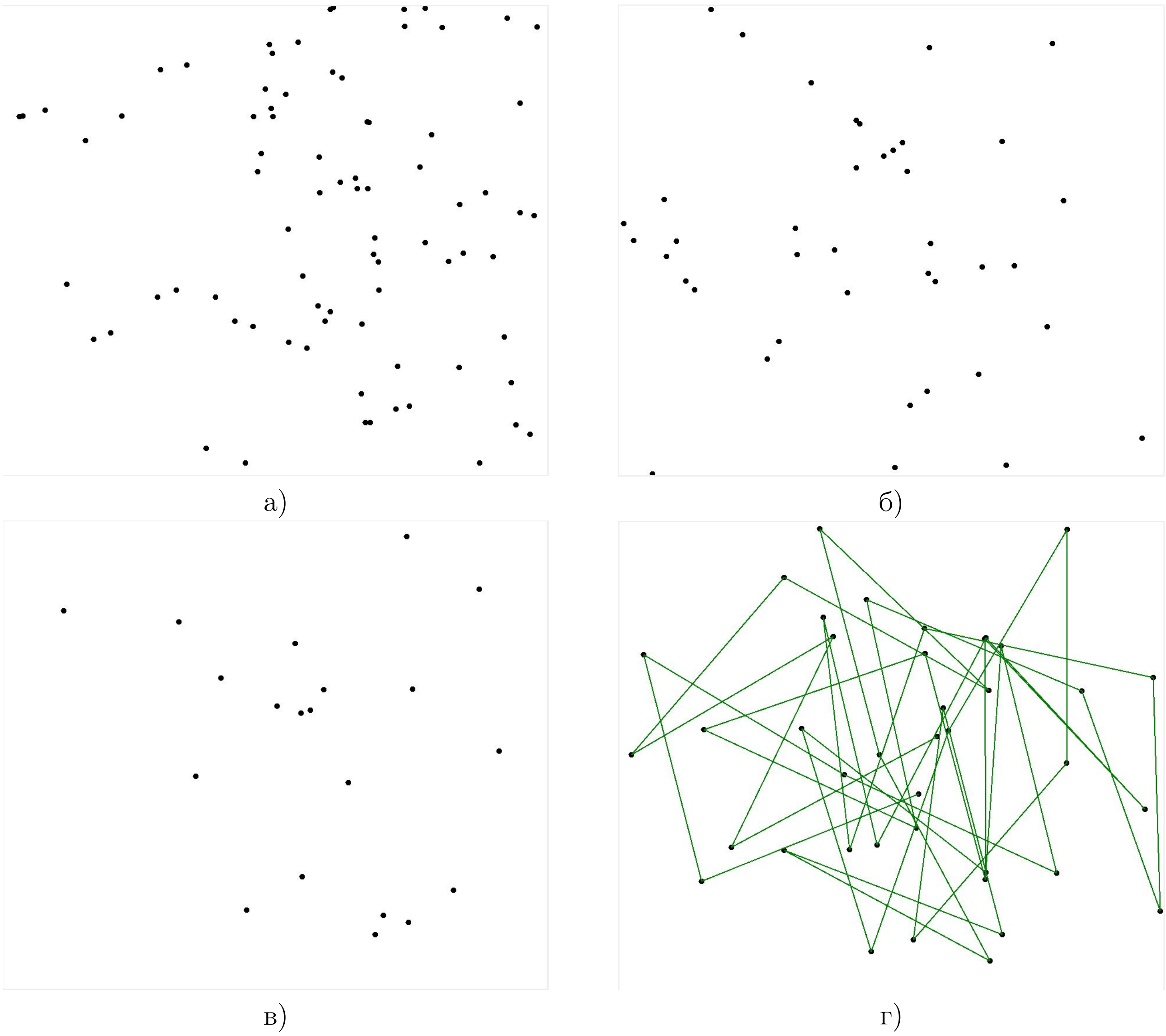
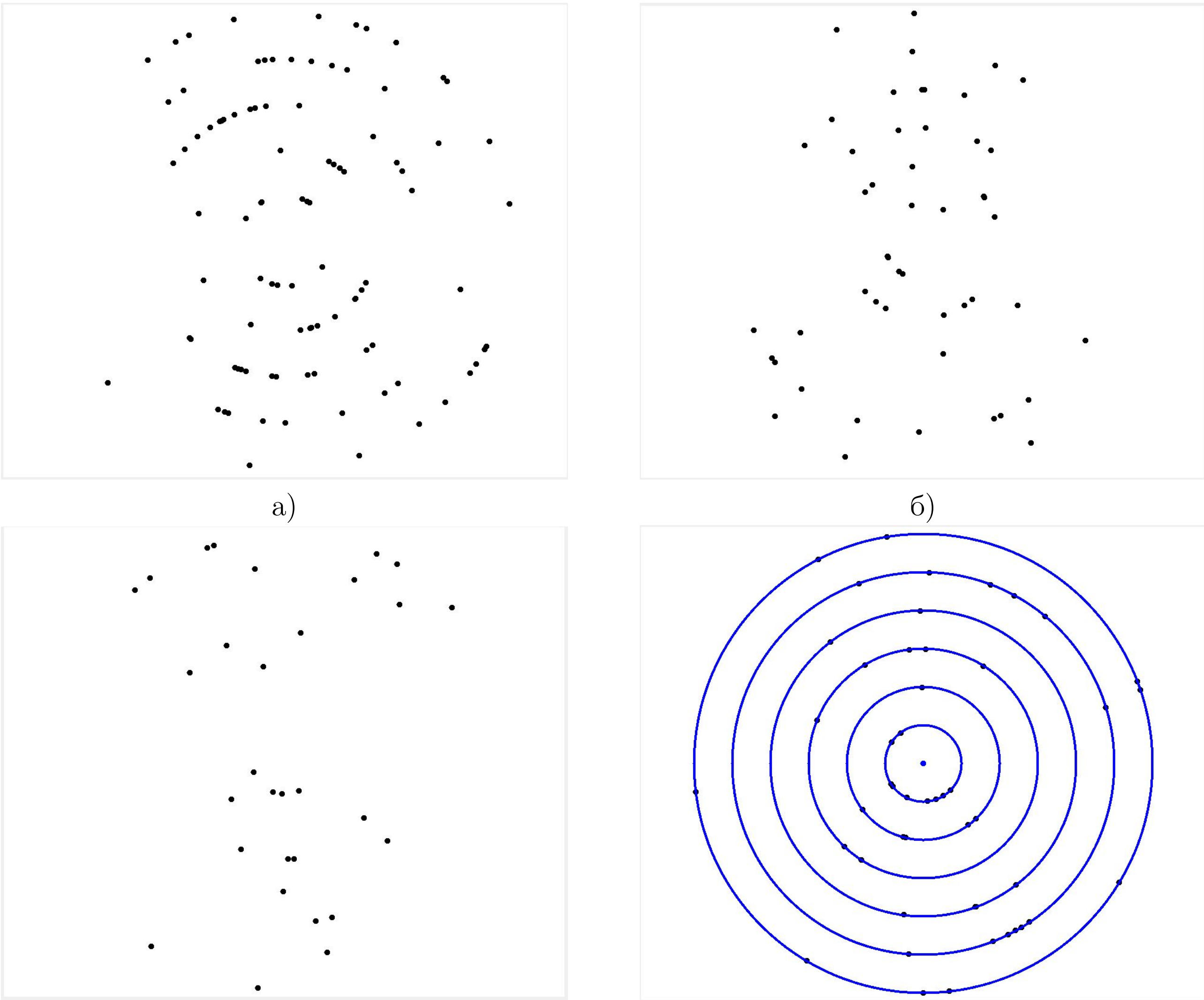


Рис. 9: Стратегия «Случайное блуждание»



а) б)

в) г)

Рис. 10: Стратегия «Концентрические окружности»