Федеральное государственное автономное учреждение
высшего профессионального образования

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ «ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Образовательная программа «Прикладная математика»
Бакалавр

О Т Ч Е Т

по проектной работе

Моделирование пространственно-распределенных игр

Выполнил: студент группы 205
Кочарян Давид рудикович

Руководитель проекта:
Доцент *Буровский Евгений Андреевич*

Москва 2021

Содержание

1 Описание задачи **3**

1.1 Игровое поле 3

1.2 Механика взаимодействия ролей 3

1.3 Взаимодействие по схеме *квадрат* 3

1.4 Взаимодействия по схеме *треугольник* 3

1.5 Механика хода 3

1.6 Цель проекта 4

2 Решение **5**

2.1 Описание параметров в поле тира *квадрат* 5

2.2 Описание параметров в поле типа *треугольник* 6

2.3 Смешанный тип взаимодействия *квадрат-треугольник* . . 6

2.4 Результаты 8

3 Описание использованных в проекте способов и технологии **9**

3.1 Программное обеспечение 9

3.2 Назначения функции 9

3.2.1 evolve 9

3.2.2 create_fields 9

3.2.3 play 9

3.2.4 nsize 10

3.2.5 see 10

3.2.6 summa 10

3.2.7 see_sr 10

Содержание

1 Описание задачи

В данное работе рассматривается взаимодействие игроков основанное на дилемме заключённого из работы [1] ,в том числе далее называемых агентами на поле размеров $L \times L$. Каждый агент имеет одну из двух ролей: кооператор или дефектор, которые имеют свой механизм работы.

1.1 Игровое поле

Игроки размещены на квадратной двухмерной решетке размеров $L \times L$. Таким образом у каждого агента восемь соседей, девять объктов для взаимодействия, включая самого себя.

1.2 Механика взаимодействия ролей

Как уже было сказано есть всего две роли кооператор и дефектор, которые могут взаимодействовать между собой. Так при взаимодействии кооператора с кооператором каждый получит по R очков, в случае если у агентов разные роли дефектор получит T очков, а кооператор P, если же оба игрока являются дефекторами оба получат S.

1.3 Взаимодействие по схеме *квадрат*

Определив, выдачу награду для ролей, остаётся задать соседей с которыми будет взаимодействовать игрок. В методе квадрата агент играет с восьмью свои соседями и самим собой, что напоминает правильный четырёх угольник 3×3.

1.4 Взаимодействия по схеме *треугольник*

Данный метод отличается от предыдущего тем, что игрок из обзора игрока исключаются два соседа по главной диагонали (верхний левый и нижний правый агенты).

1.5 Механика хода

В течение одного хода для каждого игрока будет проводиться подсчёт очков, после чего агент скопирует роль соседа-игрока с наибольшим количеством баллов, после счёт каждого участника обнуляется и цикл завершается. Во время первого запуска роли игроков генерируются случайным образом, с заданным процентным соотношением кооператоров и дефекторов.

1.6 Цель проекта

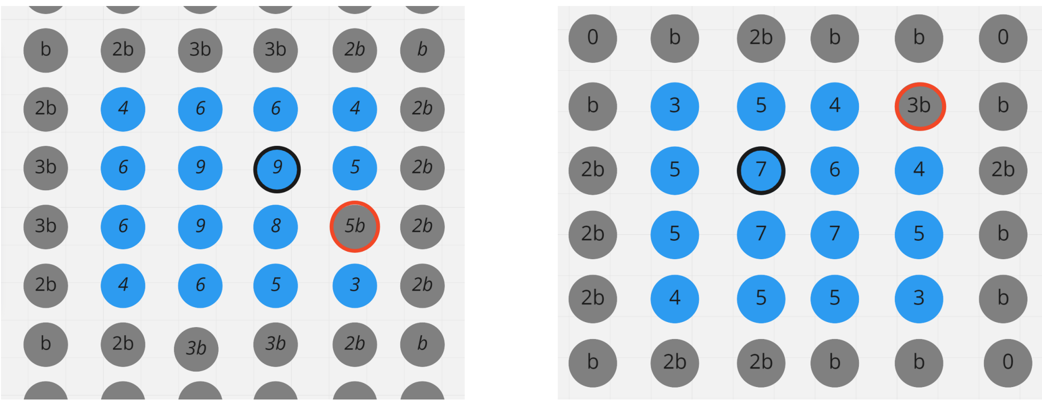
Рассмотреть плотность кооператоров в механике совмещающей оба типа взаимодействия.

2 Решение

2.1 Описание параметров в поле тира *квадрат*

Для того, чтобы ответить на поставленный вопрос необходимо рассмотреть влияние коэффициентов R,T,P,S на плотность распределение кооператоров, в данный работе $R = 1; T = b; S = 0; P = 0$. Сначала возьмём квадратную решётку.

Берём наиболее выгодный вариант для кооператоров, который является наихудшим для дефекторов, поле такого типа изображена на рисунке 2a. Итак у нас есть кооператор восемь соседей, которого имеют ту же роль, однако все остальные являются дефекторами, грубо говоря есть остров кооператоров в море дефекторов. Возникает вопрос при каких условиях агенты будут изменять свои роли на противоположные. Рассмотрим верхний правый угол квадрата кооператоров, далее именуемых C от английского слова cooperator. У этого игрока соседями для взаимодействия являются 3 C и 5 дефекторов, далее именуемые D от английского слова defector. Так соседом с ролью C и наибольшим числом очков является кооператор в центре квадрата, сумма его баллов составляет 9, у D наибольшее число очков составляет $5b$. Таким образом чтобы рассматриваемый игрок сменил роль должно выполняться условие $b \geqslant \frac{9}{5}$.



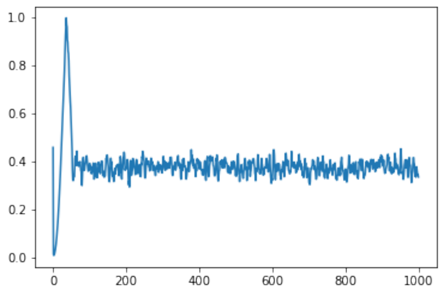
(a) Наиболее выгодное поле для кооператоров(взаимодействие типа квадрат)

(b) Наиболее выгодное положение для кооператоров(взаимодействие типа треугольник)

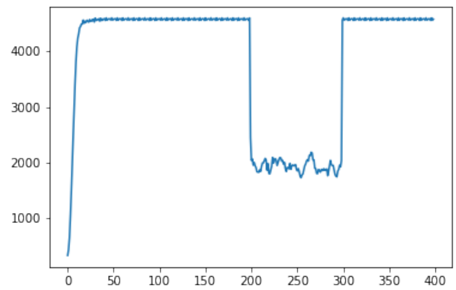
2.2 Описание параметров в поле типа *треугольник*

Перейдём к треугольной решётке. Также как и для квадрата рассмотрим наиболее выгодную для C ситуацию, она изображена на рисунке 2b. C учётом особенности треугольной решётки, наибольшее количество очков среди C имеет агент в центре кластера, среди D максимум очков равен $3b$. Так чтобы агент сменил роль $b \geqslant \frac{7}{3}$

Важно отметить, что значения функции $f_c(i)$ в один момент не покидают ограниченной области, то есть можно сказать, что есть определённая окрестность, в которой находится N_c. В статье [2] это состояние называется steady state(устойчивое состояние). На графиках



(a) График $f_c(i)$ для поля квадратов



(b) График $f_c(i)$ для поля треугольников

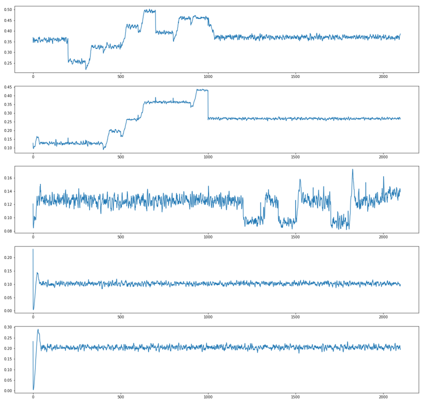
2.3 Смешанный тип взаимодействия *квадрат-треугольник*

После запуска обоих типов взаимодействия стало ясно, что есть определенное значение соотношения количества D и C на поле для различных b. Ко всему надо добавить, что хоть в обоих запусках есть конечное значение для количества, однако корреляция номера хода и N_C имеет разные формы. Введем функцию, которая будет показывать отношение N_C к N:

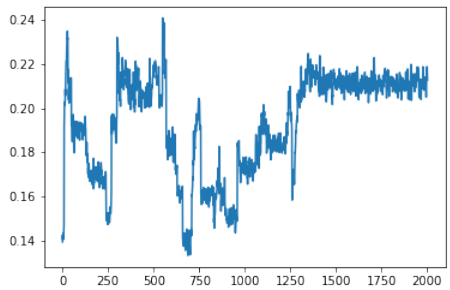
$$f_c = \frac{N_{ci}}{N}$$

Итак возникает вопрос по какой причине графики различны, даже при одинаковых начальных положениях агентов и тех же коэффициентах b.

В данной работе рассматривается случай зависимость f_c для поля агентов с обоими типа взаимодействия. Для этого, на площадке будут и квадраты и треугольник в определенном соотношений, чтобы лучшее рассмотреть зависимость функции будем обозначать количество агентов



(a) Графики $f_c(i)$ для смеси полей

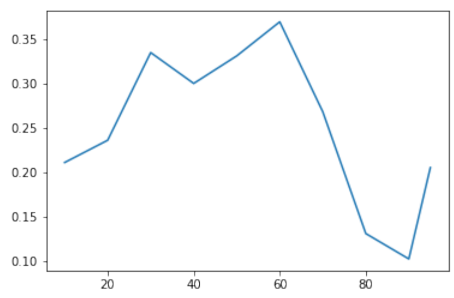


(b) Усреднённое значение $f_c(i)$ для $b = 1.9$,$t = 10$

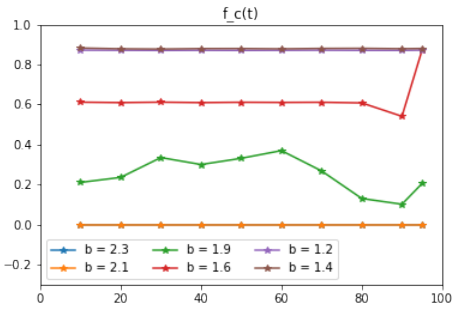
"треугольников"как t выраженное в процентах, тогда количество игроков "квадратов"будет выраженно числом k, также имеющего размерность процентов.

Перейдем к тому как будет рассматриваться зависимость. Всего взято десять значении для $t = (10, 20, 30, 40, 50, 60, 70, 80, 90, 95)$, значения $t = \{0, 100\}$, не объясняются в этой работе, так как эти случаи чистых полей "квадратов"и "треугольников"уже рассмотрены. Итак перейдём к модулированию, в среднем каждое поле будет делать 2100 шагов этого достаточно чтобы достичь устойчивого состояния, при этом для каждого t будет взято десять различных начальных положении D и C, при одинаковом b. Как результат несколько графиком можно увидеть на 3a. Далее усредним значения по графикам и получим для каждого t, на 3b изброжено усреднения только для одного блока данных. Теперь модно составить корреляцию f_c и t при $b = 1.9$, она изображена на 4a

Теперь рассмотрим зависимость f_c от b для тех же самых значении t, значения $b = \{1.2, 1.6, 1.9, 2.1, 2.3\}$. Для этого проделаем те же самые действия, что и для $b = 1.9$. Результате получен график 4b Очень легко понять почему зависимость имеет такой вид. Дело в том, что не смотря на наличие двух типов взаимодействия, положительного эффекта друг на друга она не оказывает. Чтобы понять это достаточно рассмотреть их вместе детальнее. Дело в том, что значения коэффициента b для победы дефекторов уменьшилось в случае квадратов, так как если рассмотреть наиболее выгодную позицию для C_k, то там могут появиться треугольники, которое в потенциале получает меньше очков чем те же самые квадраты на их месте, если t появиться на месте главного k, как наи-



(a) График $f_c(t)$ при $b = 1.9$



(b) График $f_c(t)$ с различными b

большее возможное значение получаемое C_k изменится на лучшее C_t. В то же самое для t в этом никаких изменении не произошло, хотя стоит сказать, что если главный C_t может быть заменён на C_k, что повысит значение b для победы дефекторов в этой локальной точке.

2.4 Результаты

Итак глядя на график 4b можно сказать несколько вещей:

-написана универсальная программа, которая позволяет не только модулировать неограниченное число разных начальных полей задавая значения для $b, t, L, steps$, но и визуализировать полученные данные.

-полученная графическая зависимость количества кооператоров при различных отношения t и k.

- нарисована зависимости $f_c(t)$ при различных b.

- $f_c(t)$ при $b = 1.9$ является хаотическим режим даже в смежном типе взаимодействия.

- C для различных b принимает меньшее значение в сравнении с полем *треугольников*, но при этом большее чем *квадраты*.

- значение $b = 2.1$, при котором начинается хаотический режим у *треугольником*, в с мешанной системе приводит к "поражению" C.

3 Описание использованных в проекте способов и технологии

3.1 Программное обеспечение

Для решения поставленной задачи был использован язык программирования python с библиотеками numpy и matplotlib. Весь код и все графики были обработаны в интерактивной оболочке для программирования Jupyter Notebook

3.2 Назначения функции

3.2.1 evolve

- обрабатывает ход.
- принимает аргументы: steps - количество шагов; b - коэффициент беспорядка; field - игровое поле в виде массива $L \times L$, где L - размер поля.
- возвращает обработанное field и количество кооператоров в виде массива n.

3.2.2 create_fields

- создает игровые поля.
- принимает аргументы: t - количество треугольников в процентах; L - размер поля; C -количество кооператоров в процентах; R - индекс для псевдослучайного генерирования чисел; kolvo - сколько полей необходимо создать.
- возвращает: area - это массив размеров $kolvo \times L \times L$ где каждый элемент area[i] это отдельное только что созданное игровое поле.

3.2.3 play

- запускает обработку поля.
- принимает аргументы: k - на каком ходу было поле при передаче в функцию; kolvo - сколько передано полей; area - поля, которые обработаются; b - коэффициент беспорядка.
- возвращает: area - обработанное поле; k - ход, на котором остановилась обработка поля; e - массив содержащий число кооператоров на полях.

3.2.4 nsize

- обновляет n в соответствии с e.
- принимает аргументы: e - массив новых данных о количестве кооператоров; k - номер нынешнего хода; n - массив количества кооператоров который необходимо обновить; steps - количество сделанных ходов.
- возвращает: -
- возвращает: n - актуальный массив с количеством кооператоров.

3.2.5 see

- визуализирует n.
- принимает аргумент: n - массив с количеством кооператоров.
- возвращает: -

3.2.6 summa

- усредняет n.
- принимает аргумент: n - массив с количеством кооператоров.
- возвращает: s - усреднённый массив с количеством кооператоров.

3.2.7 see_sr

- визуализирует массив s.
- принимает аргумент: s - массив усреднённого количества. кооператоров.
- возвращает: -

Список литературы

[1] Martin A. Nowak & Robert M. May: *Evolutionary Games and Spatial Chaos*

[2] Evgeni Burovski, Aleksandr Malyutin, Lev Shchur *On the geometric structures in evolutionary games on square and triangular lattices*

[3]

[4] Источник информации по работе с matplotlib: https://matplotlib.org

[5] Источник информации по работе с numpy: https://numpy.org