

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	2
1 ЛИТЕРАТУРНЫЙ ОБЗОР	3
1.1 Определение основных понятий	3
1.2 Эксперимент Эллсберга. Дилемма студента	4
1.3 Роль когнитивных искажений в принятии решений	7
1.4 Значимость парадоксов в принятии решений	11
1.5 Модель социально-значимых Интернет ресурсов	12
2 КВАНТОВЫЕ МЕТОДЫ ПОСТРОЕНИЯ МОДЕЛЕЙ ПРИНЯТИЯ РЕШЕНИЙ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ	19
2.1 Открытые квантовые системы	19
2.2 Принятие решения в квантовом формализме	22
3 КВАНТОВОПОДОБНАЯ МОДЕЛЬ ПРИНЯТИЯ РЕШЕНИЙ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ ВО ВЗАИМОДЕЙСТВИИ С ОКРУЖЕНИЕМ	27
3.1 Постановка задачи для квантовоподобной модели	27
3.2 Основные уравнения	31
3.3 Методы решения для немарковского резервуара	33
3.4 Моделирование внешнего информационного резервуара	39
3.5 Программное моделирование	48
ЗАКЛЮЧЕНИЕ	54
СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ	56
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	57
Приложение А. Основные аналитические методы построения модели .	60
Приложение Б. Исходный код	69

ВВЕДЕНИЕ

Работа начинается с обзора литературы, в котором рассказываются ключевые парадоксы в принятии решения индивидом в условиях неопределенности, рассматриваются существующие модели принятия решений, раскрываются необходимые понятия из квантовой физики для объяснения последующих рассуждений в основной работе. Актуальность темы заключается в разрешении парадоксов когнитивных и смежных наук в задачах принятия решений с помощью квантовых подходов. Работа заканчивается результатами численного моделирования принятия решения человеком в условиях неопределенности, находящимся в информационной среде. Актуальность работы заключается в пересмотре существующих квантово-подобных моделей принятия решений в рамках немарковских процессов поведения для учета эффектов случайности мышления. Целью работы является разработка немарковской динамической квантово-подобной модели принятия решений в условиях неопределенности. Задачи:

- Разработать модель принятия решений с учетом немарковского взаимодействия агента с информационным окружением(резервуаром)
- Исследовать влияние структуры резервуара на принятие решения агентом
- Сравнить полученные результаты с существующими моделями

В работе были использованы методы интерационного решения, методы решения линейных дифференциальных уравнений, методы дискретной математики, методы поиска аналитического решения. В работе было исследовано влияние структуры резервуара на принятие решения агентом в условиях неопределенности, были найдены граничные условия перехода разработанной модели к существующим.

1 ЛИТЕРАТУРНЫЙ ОБЗОР

1.1 Определение основных понятий

Под „решениями“ будем понимать совокупность рассматриваемых возможностей, которые выделены тем или иным способом человеком, группой лиц или внешней средой. Этой терминологии в иностранном языке соответствует слово "decision". Тогда, принятие решения - это процесс выбора определенной возможности человеком или группой лиц. Проблемой принятия решения занимаются многие дисциплины: теория принятия решений, системный анализ, теория статистических решений, информатика, искусственный интеллект, когнитивная психология, теория поведения, теория игр и др. Теория принятия решений в основном занимается разработкой различных методов и средств, необходимых человеку для формулировки вариантов решения проблемы, нахождением наилучшего решения проблемы и обоснованием выбора решения [1]. В применении к когнитивной психологии, теория принятия решений используется для изучения причин выбора определенного решения человеком. Имеются задачи, которые могут иметь комплекс различных дисциплин, например включать в себя теорию игр, когнитивную психологию и искусственный интеллект. Такие задачи обычно относят к теории принятия решений и не рассматривают отдельно друг от друга. Примером тому является эксперимент одноразовой дилеммы заключенного, о котором будет рассказано далее.

В теории принятия решений существует большое количество классификаций проблем, связанных с различными сферами развития в других науках, одним из которых является понятия риск и неопределенность. В широком смысле, "неопределенность" трактуется неоднозначно и зависит от условий решаемой задачи. Для описания неопределенности, теория принятия решения применяет, в частности, инструмент теории нечетких множеств [2]. Теория нечетких множеств позволяет отнести некоторое множество X к

некоторому подмножеству A , используя параметр степени принадлежности μ_A . Степень принадлежности является функцией и принимает значения в интервале $[0, 1]$. В формальном виде, нечеткое множество можно описать следующим образом: $A = \{(x, \mu_A(x)) | x \in X\}$.

Помимо инструментов для описания неопределенности большинства наук, в когнитивной психологии, существует большое количество концепций, по которым описывается поведение человека. Одним из таких концепций является принцип уверенности Сэвиджа. Этот принцип гласит, если положительный результат события S дает выбор решения R , отрицательный результат события \bar{S} дает выбор решения R , тогда отсутствие информации о событии S даст выбор решения R . Иначе говоря, событие S никак не влияет на выбор решения, даже если событие S неопределено. В таком понимании, неопределенность понимается как неоднозначность объективного состояния мира. Принцип уверенности Сэвиджа показывает рациональное мышление. Однако, принцип Сэвиджа нарушается в ряде экспериментов, где тому пример эксперимент с эффектом неоднозначности Элсберга [3] и эксперимент с покупкой билета [4].

1.2 Эксперимент Элсберга. Дилемма студента

Первым примером рассмотрим эффект неоднозначности Элсберга. Данный эффект формулируется в мысленном эксперименте, когда имеются две урны, в каждой из которых по 100 мячей, где в первой урне 50 красных и 50 черных, во второй соотношение неизвестно. Участникам эксперимента задается вопрос: "Какую выберете урну, чтобы достать черный мяч?". В случае угадывания испытуемый получает денежное вознаграждение, иначе ничего не теряет [3]. Для данной задачи был проведен эксперимент, большинство испытуемых были склонны выбирать первую урну, иначе говоря большинство человек решило рискнуть, поскольку вероятность вынуть черный мяч 50/50. А

неопределенность второй урны было не привлекательным, хотя соотношение внутри этой урны могло быть 70/30, 90/10 или даже 100/0, хотя при проигрыше игрок ничего не терял [3, 5, 6]. Иллюстрация этого эксперимента приведена на рисунке 1.1



Рисунок 1.1 – Эксперимент Эллсберга характеризующее когнитивное искажение, в котором принятие решения страдает из-за недостатка информации. В эксперименте предпочтительней будет первая урна, для которой вероятность благоприятного исхода известна, нежели вторая, где вероятность благоприятного исхода неизвестна.

Более простым примером неопределенности является ситуация связанности независимых событий. Например, студент плохо готов к экзамену, поэтому ему надо решить, либо идти на экзамен и попытаться защититься, либо пропустить экзамен и пойти на дополнительную попытку. При этом, студент хочет полететь на отдых, для чего необходимо купить билеты заранее. Эти два вопроса, о экзамене и покупке билетов, никак не связаны, однако, студент может быть не удовлетворен результатом экзамена и не полететь или не сдавать экзамен и спокойно полететь. Иллюстрация этих возможных вариантов событий представлена на рисунке 1.2

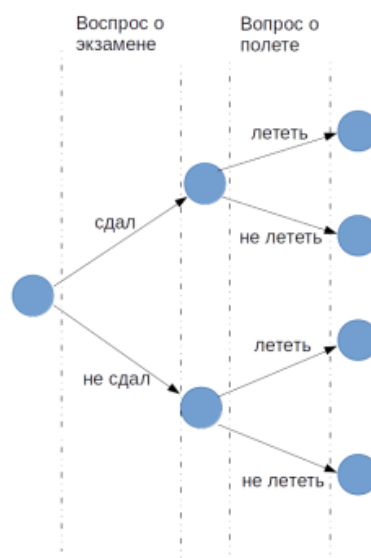


Рисунок 1.2 – Цепь возможных исходов независимых событий [7]

где кругами обозначаются определенные состояния события. Если использовать принцип уверенности Сэвиджа, то варианты событий будут выглядеть как на рисунке 1.3

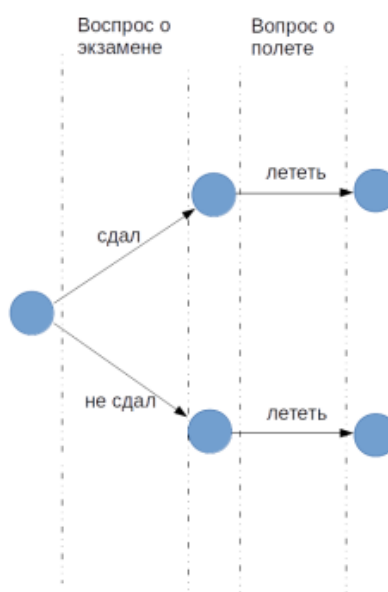


Рисунок 1.3 – Цепь возможных исходов, образованных по принципу Сэвиджа

Из этого рисунка видно, что человек всегда выберет один вариант для второго вопроса и при этом ответ на первый вопрос никак не влияет на второй вопрос. Однако, в этом эксперименте было получено, что многие люди решили купить билеты в случае провала экзамена или в случае сдачи, но решили не

покупать в случае неизвестных результатов экзамена [4].

Важным результатом этих экспериментов является выявление эффекта дизъюнкции. Эффект дизъюнкции гласит, что лучше выбрать вариант x чем y , если известно о результате события S , или лучше выбрать вариант y чем x , если о результате события S ничего не известно. Данный эффект был сформулирован в работе Шафира и Тверского [7].

1.3 Роль когнитивных искажений в принятии решений

Еще один пример связан с ошибкой конъюнкции, который был получен в эксперименте Тверского и Канемана [8]. Эксперимент показывает, что для человека некоторые составные факты могут быть весомее общих. Для постановки эксперимента, каждому участнику дается информация о Линде. "Ей 31 год, она одинокая, искренняя и очень яркая. Она училась на факультете философии. Будучи студенткой была обеспокоена вопросами дискриминации и социальной несправедливости, участвовала в демонстрациях против ядерного оружия". После задавался вопрос: "Какой факт о Линде более вероятен?"

- "Линда - кассир в банке";
- "Линда - кассир в банке и активная феминистка";

Для эксперимента были отобраны 142 студента магистратуры. Для одной половины студентов порядок альтернатив был прямой, для других обратный, но эта манипуляция не принесла других результатов. В результате 85% респондентов выбрали второй вариант: "Линда - кассир в банке и активная феминистка" [9].

Эти эффекты показывают, что люди не оценивают все ветви соответствующего дерева решений, даже если количество результатов велико.

Помимо принятия решения человеком в условиях неопределенности событий, в когнитивной психологии рассматриваются задачи в условиях

неопределенности результата решения других людей. Ярким примером такой неопределенности является дилемма заключенного. Разница между предыдущими экспериментами заключается в раскрытии большей информации о неопределенности. Другими словами, респонденту будет известна предположительная информация о выборе другого человека. Таким образом, предполагается раскрыть больше информации об когнитивных эффектах происходящих в условиях неопределенности.

Дилемма заключенного является основной задачей теории игр и уже имеет большое количество направлений исследования. Самым интересным направлением стоит отметить исследования повторяющихся игр, то есть когда дилемма заключенного повторяется по несколько раз с теми же игроками. Но не менее интересным стоит отметить игры на один раз в контексте когнитивной психологии. Разница между многократной и одиночной игрой заключается в степени неопределенности. Человек абсолютно не знает о другом человеке в первую игру и не может определить тактику другого человека. При многократном повторении игроки могут угадывать поведение другого и таким образом неопределенность начинает исчезать.

Но что будет, если игрок будет знать о другом предположительную информацию. В этом случае результат для первых экспериментов был неожиданным, в них наблюдались эффект неоднозначности и эффект дизъюнкции. Для объяснения всей сути проблемы необходимо начать с постановки задачи.

В игре участвуют двое заключенных, каждого из которых поймали во время преступления. Каждому заключенному дается выбор либо предать другого, либо молчать. Их общий выбор будет влиять на величину наказания для обоих согласно таблице 1:

Таблица 1 – Таблица выплат для классической задачи

		заключенный Б	
		молчать	предать
заключенный А	молчать	А - 0.5 года Б - 0.5 года	А - 10 лет Б - свободен
	предать	А - свободен Б - 10 лет	А - 2 года Б - 2 года

Каждый заключенный знает про эту таблицу и пытается свести результат к минимальному возможному наказанию. Согласно теории игр, каждому заключенному по Парето-оптимуму необходимо выбирать вариант молчать.

Парето-оптимум позволяет рассматривать задачу для обоих заключенных, позволяет найти общий выигрыш для двух заключенных. Рассмотрение задачи для каждого в отдельности позволяет прийти к выводу, что самым лучшим вариантом является предательство. Поскольку в случае предательства одного заключенного, промолчавший заключенный получает максимальное наказание. Случай с предательством обоими заключенными является равновесием Нэша. Согласно равновесию Нэша, существует стратегия или набор стратегий, изменив которые невозможно получить больший выигрыш, если другие игроки не поменяли свои стратегии. Такой стратегией является предательство обоих заключенных, но такая стратегия не является Парето-оптимальным. Конечно, если заключенные знают друг друга давно и они связаны "воровским законом" или игра является многократной, то их стратегия оказывается молчание и таким образом равновесие Нэша приближается к Парето-оптимуму. Но стратегия молчания не работает для других ситуаций, в том числе для однократных игр, где игроки не знают друг о друге [10].

Для экспериментов с людьми удобнее пользоваться другой таблицей выплат, нежели ранее представленной в таблице 1, которая будет больше

интересовать игроков в выигрыше. В одной из работ была представлена следующая таблица выплат 2. В эксперименте испытуемыми были 80 студентов

Таблица 2 – Таблица выплат в эксперименте

		игрок Б	
		сотрудничать	предать
игрок А	сотрудничать	А - 75 баллов Б - 75 баллов	А - 25 баллов Б - 85 баллов
	предать	А - 85 баллов Б - 25 баллов	А - 30 баллов Б - 30 баллов

Принстонского университета. Каждому студенту предоставлялось по 40 игр из которых 6 были с дилеммой. Другие игры были необходимы для устранения возможности повтора игроком своей предыдущей стратегии. После прохождения всех игр у каждого участника подсчитывалось количество баллов, в которые также входили результаты игры с дилеммой (в среднем выплачивалось по 6 долларов). Игра была разделена на 3 ситуации, где в первой ситуации обыгрывалась стандартная дилемма заключенного, во второй ситуации игроку предоставлялась информация о сотрудничестве другого игрока, в третьей ситуации игроку предоставлялась информация о предательстве другого игрока. В результате такой игры получились результаты представленные в таблице 3

Таблица 3 – Результаты эксперимента дилеммы заключенного

стандартная игра	предать	63%
	сотрудничать	37%
другой игрок сотрудничает	предать	84%
	сотрудничать	16%
другой игрок предает	предать	97%
	сотрудничать	3%

Как видно из этой таблицы, большинство придерживается стратегии предательства. Также можно заметить, что некоторые придерживаются стратегии "око за око в момент когда человек узнает стратегию другого игрока. Сравнивая результаты стандартной игры и с предоставлением информации игроку, можно заметить, что большее количество игроков склонно сотрудничать в отсутствие информации о результате выбора другого игрока (37%), чем в момент присутствия (16% и 3%) [7]. Такое поведение полностью соответствует эффекту дизъюнкции, но можно свести к ошибке конъюнкции, если принять тот факт, что среднее число вероятности с известным выбором другого (16% и 3%) оказывается меньше вероятности стандартной игры (37%).

1.4 Значимость парадоксов в принятии решений

Как видно из этого эксперимента, в условиях неопределенности испытуемые в стандартной дилемме заключенного более склонны к сотрудничеству, чем в определенных условиях. Все эти эксперименты, перечисленные ранее представляют основу в данной работе, поскольку с этих работ начинается развитие квантово-подобных моделей. Стоит отметить, что выводы из экспериментов применимы не только к ситуациям с оппонентом с какой-то определенной проблемой, но и с другими социальными, экономическими и когнитивными ситуациями. Так например, проблема дилеммы заключенного применима так же к отношениям между обществом и человеком.

Стоит отметить, что объяснение всех этих эффектов проводились задолго попыток применения квантового формализма. Эффект ошибки конъюнкции экспериментаторы Канеман и Тверски пытались объяснить с помощью эвристической репрезентативности. Однако такой подход является неформальным и плохо определенным, а также учитывают только ограниченные случаи [11–13]. Существует предположение, что люди склонны оценивать

вероятность конъюнкции из некоторой комбинации вероятностей компонентов, но эмпирических подтверждений этому предположению не существует [14]. Другое предположение основано на понятии индуктивного подтверждения, так же как определено в байесовской теории, и дают для этого экспериментальное обоснование. Существуют утверждения, что в рамках байесовской системы существуют случаи когда это не является ошибкой и может быть учтена рационально [15, 16]. Конечно, каждый конкретный случай можно смоделировать с помощью простой и логичной модели, но часто бывают случаи когда модель не применима к другим ситуациям с другими эффектами. В этом случае формализм квантовой-механики способен соединить различные логические эффекты, которые непосредственно связаны с природой квантовой механики, в одну или несколько связанных моделей. И таким образом может дать более полное представление о когнитивных особенностях мышления человека.

1.5 Модель социально-значимых Интернет ресурсов

Прежде чем переходить к рассмотрению квантово-подобных моделей принятия решений в условиях неопределенности, необходимо рассмотреть классические модели, не использующие квантовый формализм. Особенность следующей модели заключается в том, что ее модель схожа с квантовыми, но не рассматривает остальные социальные задачи, из-за чего данная модель ограничена в применении. Это модель социально-значимых Интернет ресурсов, разработанная на основе результатов исследований в социальной психологии, в механизме запоминания информации человеком, а также на основе теории информации [17]. Данная модель в равной степени адекватно описывает поведение участников сети Интернет ресурсов.

Существует большое количество различных социальных Интернет-ресурсов. И в первую очередь это различные социальные сети,

благодаря которым человек получает персонифицированную информацию как о себе, так и о других участниках. Это блоги, благодаря которым человек получает более развернутую информацию. Различные форумы, которые позволяют обсуждать некоторую тему и др. Для предоставления участникам ресурса возможности общения, во многих Интернет-ресурсах требуется предварительная регистрация. Определенного шаблона для составления анкеты на этапе регистрации не существует, поскольку различные Интернет-ресурсы требуют не все данные о пользователе. Однако, существует некоторая тенденция в составлении анкет во время регистрации, благодаря чему можно спроектировать модель аккаунта пользователя.

В модели социально-значимых Интернет-ресурсов предлагается модель пользовательского аккаунта. Эта модель описывает большое количество различных параметров анкеты пользователя. Эта модель имеет следующий вид:

$$A = \langle Pers, Cont, CoC, Prof \rangle, \quad (1)$$

где *Pers* — подмножество идентифицирующей информации, *Cont* — подмножество контактных данных, *CoC* — подмножество данных о социальных связях, *Prof* — подмножество различной личной информации.

Помимо этого, для этой модели существует модель распространения информации, которая имеет вид:

$$I = \langle Pers, M, T \rangle, \quad (2)$$

где *Pers* — подмножество идентифицирующей информации, *M* — мнения, комментарии, *T* — отметка времени, $K = f(M)$ — когниция, элемент знания (данные, усвоенные сознанием).

Помимо этого, модель пользовательского аккаунта имеет информацию о связях с другими аккаунтами, которая позволяет рассчитать параметр

интенсивности отношений:

$$\alpha_{ik} = N \log_2 \frac{N_{ik}}{N_0}, \quad (3)$$

где N_{ik} – количество переданных сообщений/знаков между членами i и k , N_0 – среднее значение переданных сообщений в сообществе, N – некоторая константа, различная для каждого индивида.

На основе формулы 3, а также модели пользовательского аккаунта и модели распространения информации приводится схема взаимодействия пользователей и информации в виде графа (см. рисунок 1.4).

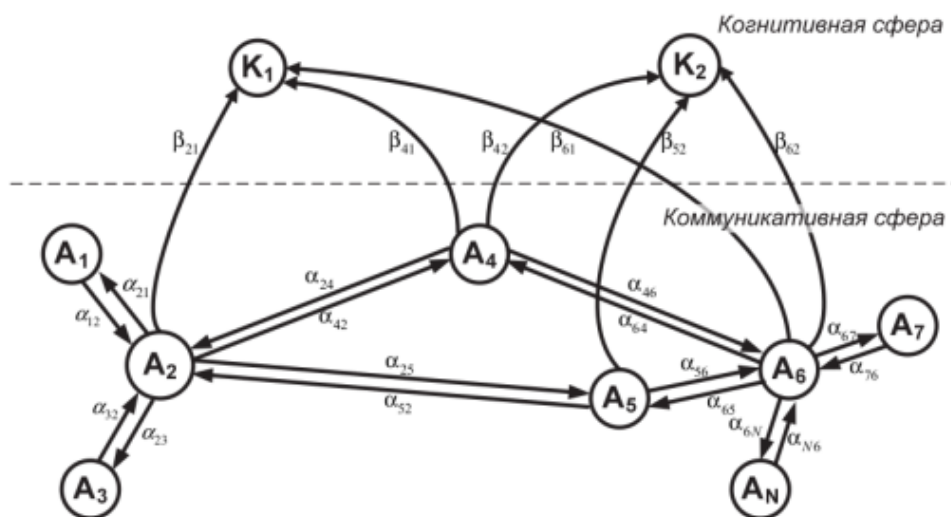


Рисунок 1.4 – Схема системы психологических отношений в социуме (социально значимых Интернет-ресурсах) [17]

На этом рисунке $A = \{A_1, A_2, A_3, \dots, A_N\}$ – это члены сети, $K = \{K_1, K_2, K_3, \dots, K_N\}$ – это когнции. Схема построена с учетом теории когнитивного диссонанса Л. Фестингера, теории структурного баланса Ф. Хайдера, теории коммуникационных актов Т. Ньюкома и теории конгруэнтности Ч. Осгуда и П. Танненбаума, для которых базовыми постулатами являются предположения о возможности получения человеком информации, его переработке, его усвоению, о стремлении человека к наибольшей связности имеющихся знаний, которые могут побуждать к действиям.

Самым простым элементом отношений в этой схеме является рассмотрение двух членов социума (О,S) и одной когниции (К). Смена значений этих отношений называется как информационно-психологическая акция (ИПА). Данная ИПА представлена на Рисунке 1.5.

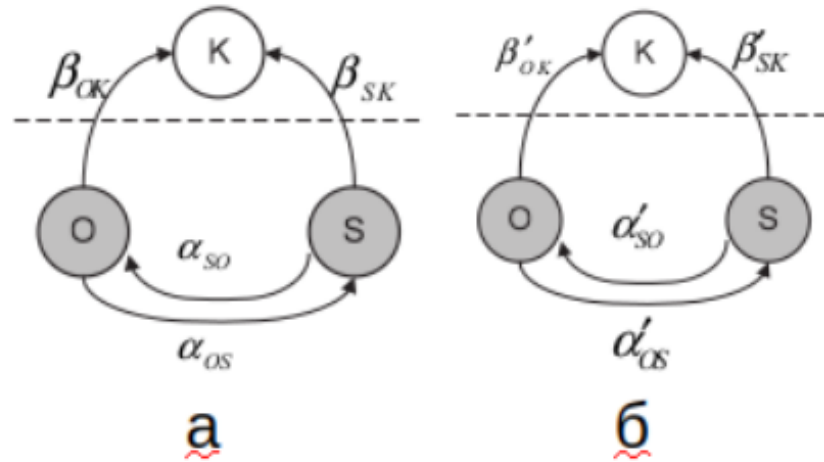


Рисунок 1.5 – Схема отношений объекта О и субъекта S друг к другу и к утверждению К: а) для события до ИПА; б) после ИПА [17]

На рисунке 1.5 а параметры α_{OS}, α_{SO} показывают отношение объекта и субъекта друг к другу, параметры β_{OK}, β_{SK} показывают отношение членов социума к утверждению. На рисунке 1.5 б, эти отношения имеют новые значения и обозначаются $\alpha'_{OS}, \alpha'_{SO}, \beta'_{OK}, \beta'_{SK}$. Изменение значений для параметров отношений вычисляются на основе дифференциалов Осгуда и для этой модели записываются следующим образом:

$$\Delta_{OK} = \frac{|\alpha_{OS}|}{|\beta_{OK}| + |\alpha_{OS}|} |\beta_{OK} - \alpha_{OS}|, \Delta_{OS} = \frac{|\beta_{OK}|}{|\beta_{OK}| + |\alpha_{OS}|} |\beta_{OK} - \alpha_{OS}| \quad (4)$$

где Δ_{OK}, Δ_{OS} – это приращение к β_{OK}, α_{OS} соответственно. Различные ситуации взаимоотношений между элементами О,S,K были рассмотрены в теории структурного баланса Ф. Хайдера, а изменение отношений рассмотрено в теории коммуникативных актов Т. Ньюкомба, благодаря чему известны формулы для $\beta'_{OK}, \alpha'_{OS}$, которые получаются от начальных параметров отношений β_{OK}, α_{OS} и приращений к ним Δ_{OK}, Δ_{OS} .

В работах по моделированию информационно-психологических отношений наблюдается явление, при котором элемент социума при получения некоторой когниции распространяет его вторично. Формула, по которой это вторичное распространение информации описывается имеет вид:

$$\beta_{ij}(t) = \beta_{ij}(t_0) \cos(\rho_i \sqrt{t - t_0}) e^{-\lambda_i(t-t_0)}, \quad (5)$$

где t_0 – время последнего изменения отношения элемента социума к когниции (утверждению), t – текущее время изменения, λ_i – коэффициент «старения и утраты уникальности утверждения», ρ_i – коэффициент когнитивного «резонанса», вторичного распространения информации. График данной функции представлен на рисунке 1.6

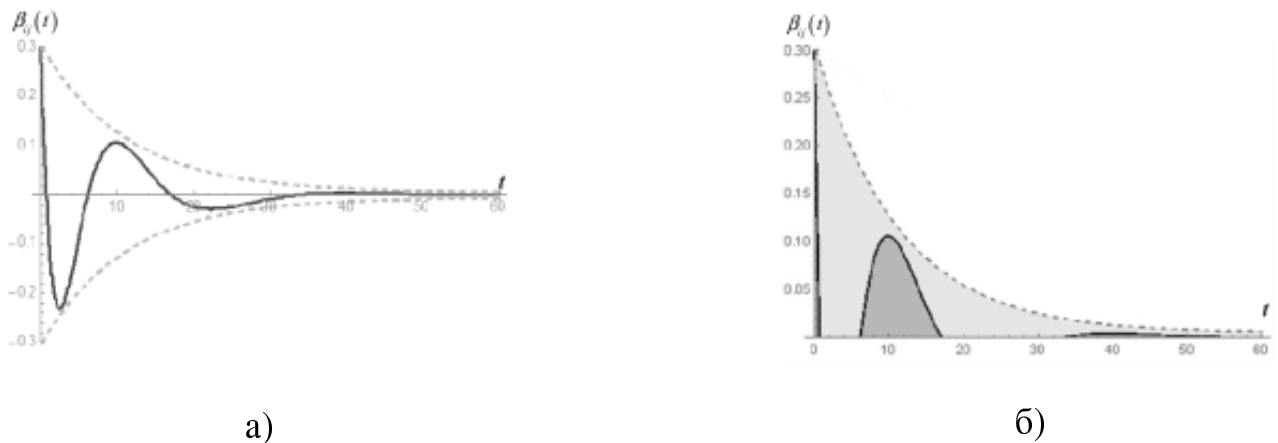


Рисунок 1.6 – Уровень отношения $\beta_{ij}(t)$ члена сети A_i к утверждению K_j [17]

Данный график, к сожалению, не показывает других зависимостей этой функции от других параметров, но на нем хорошо видно как ведет функция себя при постоянных параметрах. На рисунке 1.6а показан полный вариант с положительными и отрицательными полупериодами. Отрицательные полупериоды показывают маловероятное проявление когнитивной активности. Положительные полупериоды показывает наиболее вероятное проявление когнитивной активности. Для более детального рассмотрения графика функции $\beta_{ij}(t)$, на рисунке 1.7 представлены зависимости с дополнительным параметром (λ_i).

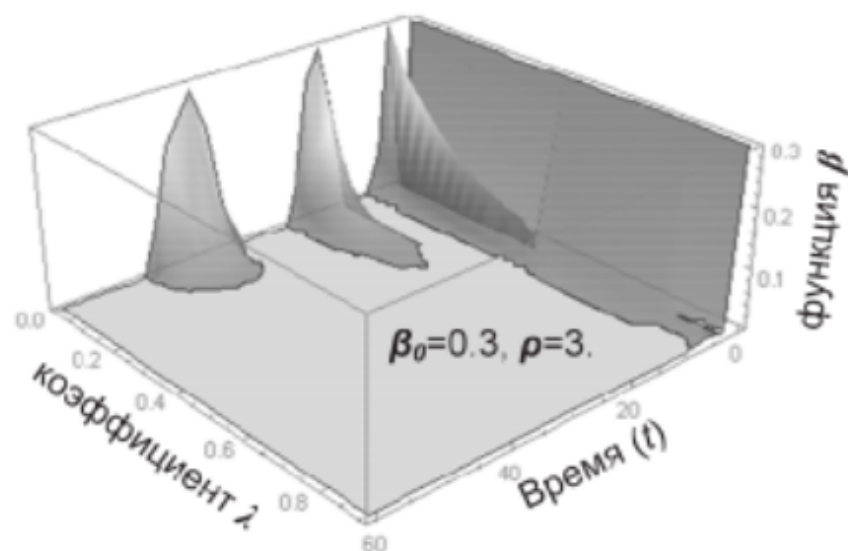


Рисунок 1.7 – График функции когнитивной активности $\beta_{ij}(t)$ в зависимости от динамики параметра λ_i [17]

Как видно из этого и предыдущего графиков, что существуют семейства монотонно убывающих зависимостей Эббингауза при определенных значениях параметра λ_i . Однако, как подчеркнули сами авторы, данную модель невозможно использовать в практической деятельности, но предлагают ограничение этой модели в определенных коэффициентах для определенных однородных групп, иначе говоря групп состоящих из людей с одним уровнем образования или предпочтений. Таким образом диапазон модели для $4.5 < \rho_i < 5$ представлен на рисунке 1.8

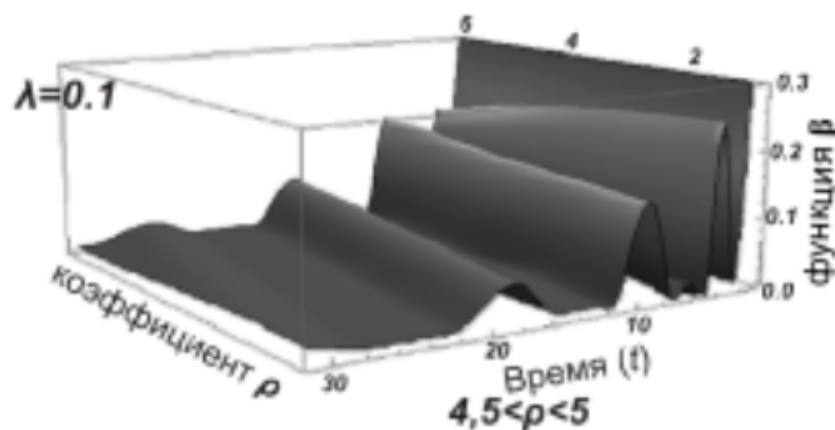


Рисунок 1.8 – График функции диффузной активности $\beta_{ij}(t)$ в зависимости от динамики параметра ρ_i [17]

В работе утверждается, что данное поведение похоже на социальную диффузию, которое представляется как „когнитивный шум“. Интересно также то, что данный шум показывает момент перехода мнения человека к мнению группы, что так же подтверждает справедливость уменьшения диапазонов ρ_i и λ_i .

Теперь рассмотрим эту проблему с точки зрения известных квантово-подобных моделей.

2 КВАНТОВЫЕ МЕТОДЫ ПОСТРОЕНИЯ МОДЕЛЕЙ ПРИНЯТИЯ РЕШЕНИЙ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ

2.1 Открытые квантовые системы

Это такая система, которая может обмениваться информацией с внешней средой. В некотором смысле, любая квантовая система является открытой, поскольку на нее могут влиять извне. Эти воздействия извне создают квантовые шумы, которые мешают рассматривать квантовые системы в простом случае. Открытые квантовые системы рассматривают ситуацию, когда имеется резервуар и система. Система может иметь чистое состояние или смешанное и разница состоит в том, что чистое состояние можно описать как векторами, так и матрицей плотности, а смешанное только матрицей плотности. Соответственно, динамику для смешанного и для чистого состояния можно задать с помощью следующего уравнения:

$$\frac{d\rho}{dt} = -i[H, \rho], \quad (6)$$

где H - гамильтониан, ρ - матрица плотности. Но уравнение (6) не описывает динамику с учетом резервуара, поэтому появилось уравнение Горини-Коссаковского-Сударшана-Линдблада (ГКСЛ):

$$\frac{d\rho}{dt} = -i[H, \rho] + \sum_j (C_j \rho C_j^\dagger - \frac{1}{2} C_j^\dagger C_j \rho - \frac{1}{2} \rho C_j C_j^\dagger), \quad (7)$$

где C_j и C_j^\dagger - некие операторы анигиляции и рождения, соответственно. Это уравнение описывает открытые квантовые системы, которые наблюдаются в экспериментах. Но это уравнение не единственное, которое может описывать открытые квантовые системы, поскольку квантовые системы могут иметь разные квантовые состояния, которые зависят от резервуара. А действие резервуара на систему может иметь марковский или немарковский характер.

Марковские процессы — это, в упрощенном понимании, процесс не зависящий от предыдущих состояний системы, который этот процесс

реализует. Иначе говоря, марковский процесс не имеет памяти и каждое новое состояние, если оно конечно, равновероятно может произойти в следующий момент времени. Самыми известными примерами марковских процессов является математическое описание броуновского движения в газах или случайное блуждание. Броуновское движение наблюдается при перемещении микроскопических объектов в газах, а случайное блуждание описывает математическую модель случайных изменений в траектории движения точки на плоскости. Немарковские процессы — это процессы, в упрощенном понимании, зависящие от предыдущих состояний системы. Самыми известными примерами такого поведения системы можно считать броуновское движение в жидкости, когда при перемещении микрочастицы жидкость начинает передавать импульс другим частицам и создавать завихрения, которые могут воздействовать и на частицу. А так же фликер-шум, который наблюдается в электрических сетях. Примеры немарковских процессов присутствуют так же и в социальных сетях, так например существует модель социально значимых интернет-ресурсов.

Помимо социальных моделей имеются другие примеры немарковских открытых квантовых систем. В одном из таких примеров рассматривалось немонотонное изменение параметра расстояния следа $D(\rho_S^1, \rho_S^1)$, в случае пары состояний ρ_S^1 и ρ_S^2 . Этот параметр характеризует различимость квантовых состояний. Суть эксперимента состоит в том, что Алиса передает свои начальные состояния открытой квантовой системы, связанной с окружающей средой, через зашумленный квантовый канал Бобу. Из-за взаимодействия среды на канал, этот параметр начинает либо уменьшаться, либо увеличиваться со временем, что означает либо потерю информации в окружающей среде, либо поступление информации из окружающей среды [18]. Данный пример схематически показан на рисунке 2.1. В тот момент, когда система просто теряет информацию, имеет случай марковского процесса, а когда система получает информацию, имеет случай немарковского процесса. Получение

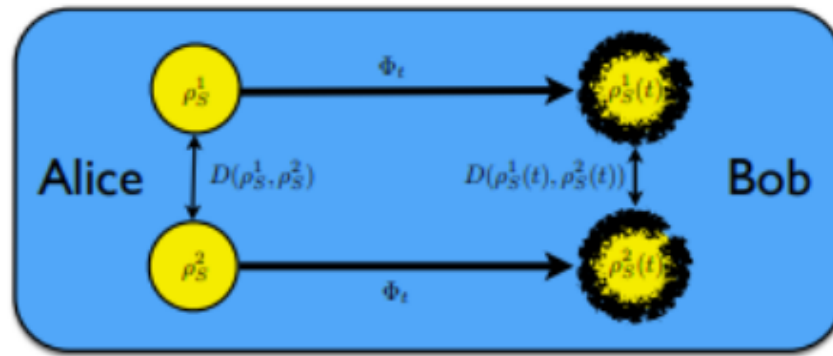


Рисунок 2.1 – Схема передачи состояния от Алисы к Бобу. Алиса готовит два различных состояния ρ_S^1 и ρ_S^2 , которая передается по динамической карте с шумом Φ_t , таким образом, сокращая доступную информацию Бобу. Потеря различимости количественно определяется расстоянием следа $D(\rho_S^1, \rho_S^1)$. [18]

информации связано с откликом окружающей среды на воздействие открытой квантовой системы, что свидетельствует о наличии памяти у окружающей среды.

Примеры немарковких процессов присутствуют так же и в оптических процессах. Частный пример тому связан с квантовым броуновским движением в оптическом резонаторе, который связан с тепловым резервуаром. В этом эксперименте установка имеет лазерный луч разделенный на сигнальный и на генераторный. Генераторный луч попадает на микромеханический резонатор, который влияет на его биение. Это биение влияет на фазу сигнального луча, который затем попадает на два фотодиода, которые регистрируют изменение фазы [19]. Биение микрорезонатора сопровождается остаточными осцилляциями, которые могут служить «памятью» процесса. Этот пример приведен на рисунке 2.2.

Результаты эксперимента показывают неомическую спектральную плотность и проявляют немарковский процесс [19].

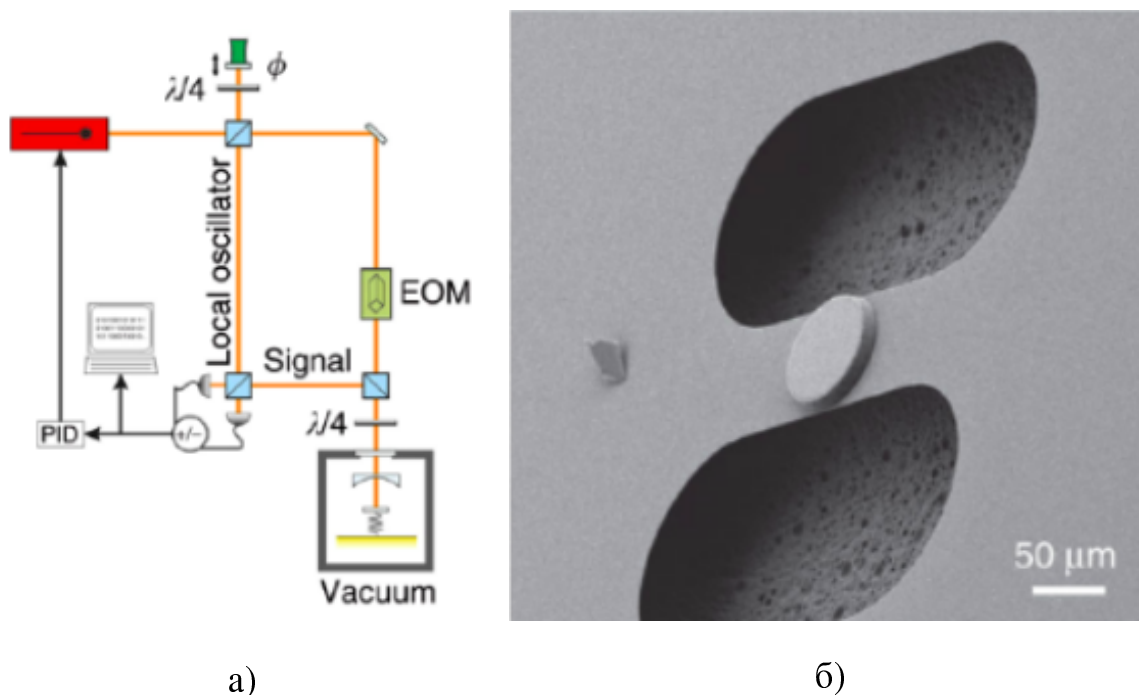


Рисунок 2.2 – а) Экспериментальная установка, состоящая из разделенного на сигнальный и генераторный лазерного излучения. Сигнальный луч получает фазу из-за движения механического резонатора. б) Сканирующая электронная микроскопия изображения тестируемого устройства [19]

2.2 Принятие решения в квантовом формализме

В процессе принятия решения на человека может воздействовать окружающая обстановка (внешние возбудители). Окружающая обстановка влияет на человека через органы чувств: зрение, слух, обоняние, физические воздействия. Органы чувств передают информацию об окружающей обстановке в головной мозг, где в дальнейшем информация обрабатывается и сохраняется. Эта информация в дальнейшем может участвовать в процессе мышления и иметь случайный характер, благодаря чему принятие определенного решения может оказаться непредсказуемым. Такое непредсказуемое решение ранее было рассмотрено в дилемме заключенного с наблюдаемым эффектом дизъюнкции. В физике эффект дизъюнкции присутствует в явлениях интерференции. В том и в другом случае наблюдается общий вклад каких-либо вариантов событий. Таким образом можно рассматривать когнитивный парадокс в дилемме заключенного

как явление интерференции в физике. Подобно этому можно привести сравнение человека как квантовой системы (атома), а окружающего его возбудителя как резервуар (гармонические осцилляторы), согласно теории открытых квантовых систем. При этом энергия спонтанного излучения атома, поглощенная резервуаром, превращается в осцилляции резервуара, которые могут иметь поведение как марковского так и немарковского процесса. Мода спонтанного излучения атома в резервуаре описывается гамильтонианом:

$$H_a = \hbar\omega a^\dagger a \quad (8)$$

где a, a^\dagger - операторы рождения уничтожения моды, \hbar - постоянная Планка, ω - частота моды излучения. Поглощения резервуаром обусловлены связью квантовой системы и резервуара, поэтому если энергия от квантовой системы передается резервуару, то через некоторое время оно за счет связи может вернуться. Такой же эффект можно наблюдать в классической физике, когда слабая связь между двумя классическими осцилляторами позволяет усиливать частоту одного осциллятора за счет другого. Марковский процесс проявляется в случае потери связи квантовой системы с резервуаром, то есть атом не получает энергию обратно от резервуара. Немарковский процесс проявляется в случае влияния резервуара на квантовую систему через связь, то есть атом получает энергию обратно от резервуара. Осцилляторы совершают упругие колебания и таким образом их можно представить как элементы некоторой кристаллической решетки [20]. Эти упругие колебания можно проквантовать также как электромагнитные колебания в диэлектрике, которые описываются статистикой Бозе-Эйнштейна и называются фононами. Гамильтониан для фононов имеет вид:

$$H_b = \sum_j \hbar\omega_j b_j^\dagger b_j \quad (9)$$

где b, b^\dagger - операторы рождения уничтожения фононов(колебаний резервуара), \hbar - постоянная Планка, ω_j - частоты фононов резервуара. В случае если

частоты фононов резервуара расположены очень плотно друг другу, то сумму можно заменить на знак интеграла. Если твердое тело (диэлектрик) находится в тепловом равновесии, то его распределение атомов по энергиям описывается уравнением Больцмана. В таком случае множество осцилляторов описываются оператором плотности в момент времени $t = 0$:

$$\rho_b(0) = \prod_{j=1}^N \rho_j(0) \quad (10)$$

где N - число атомов в резервуаре, $\rho_j(0) = (1 - \exp(-\lambda_j))\exp(-\lambda b_j^\dagger b_j)$, $\lambda_j = \frac{\hbar\omega_j}{kT}$. Таким образом гармонические осцилляторы, которые находятся в тепловом равновесии с резервуаром температурой T , аналогичны поведению фононам [20]. Общий гамильтониан квантовой системы и системы осцилляторов записывается в следующем виде:

$$H_0 = \hbar\omega a^\dagger a + \sum_j \hbar\omega_j b_j^\dagger b_j \quad (11)$$

Этот гамильтониан описывает независимые, несвязанные системы резервуара и атома. Для учета связи необходимо внести гамильтониан взаимодействия подсистем. Для ионных кристаллов такое взаимодействие обусловлено передачей электромагнитного поля. Гамильтониан взаимодействия, который сохраняет полную энергию системы, описывается следующим образом:

$$H_1 = \sum_j \hbar(k_j a^\dagger b_j + k_j^* a b_j^\dagger) \quad (12)$$

где k_j - коэффициенты связи. Обычно коэффициенты связи считаются малыми по сравнению с частотой моды излучения и упругих колебаний резервуара, поскольку зависят от характера взаимодействия систем [20]. Введенные параметры очень сильно влияют на связь между осцилляторами резервуара и атома, если частота упругих колебаний резервуара приблизительно равна частоте моды излучения. Гамильтониан взаимодействия не содержит члены

вида: $k_j a^\dagger b_j^\dagger, k_j^* a b_j$, поскольку их вклад во взаимодействие равен нулю. В итоге имеем полный гамильтониан, который описывает взаимодействие моды излучения и осцилляторами(фононами):

$$H = H_0 + H_1 = \hbar\omega a^\dagger a + \sum_j \hbar\omega_j b_j^\dagger b_j + \sum_j \hbar(k_j a^\dagger b_j + k_j^* a b_j^\dagger) \quad (13)$$

Для дилеммы заключенного каждая квантовая модель рассматривается в гильбертовом пространстве, которое может быть бесконечномерным. Для системы из двух игроков достаточно конечного пространства. Определение размерности зависит от количества исходов игры и для этого необходимо определить какие есть варианты развития событий. В задаче дилеммы заключенного участвуют двое игроков G_1 и G_2 , у каждого из которых есть по два возможных варианта выбора это сдать и не сдавать, обозначим как вектора в гильбертовом пространстве $|0\rangle$ и $|1\rangle$, соответственно. Тогда, функцию состояния каждого игрока можно описать с помощью следующей формулы:

$$|\psi\rangle_j = \sum_{j=0,1} a_j |j\rangle \quad (14)$$

где a_j - коэффициент проекции вектора $|j\rangle$.

Такую функцию можно представить с помощью двумерного гильбертова пространства \mathbb{C}^2 . Квадрат модуля коэффициента $|a_j|^2$ имеет смысл как вероятность выбора одного из векторов $|0\rangle$ или $|1\rangle$. В таком пространстве базисы векторов двух игроков могут быть повернуты на угол друг относительно друга, а значит в таком случае при выборе одинакового значения двух игроков, направление вектора состояния будет различаться из-за различного ментального восприятия мира. При использовании данной формулы можно определить вариант выбора одного игрока, однако на основе выбора первого невозможно определить выбор другого, иначе говоря их решение не влияет на результат другого, что в данной дилемме не всегда так, поскольку при повторении игры прошлый опыт влияет на обоих

игроков одновременно. Поэтому необходимо определить другую функцию состояния, которая впоследствии позволит правильно определить размерность гильбертова пространства. Поскольку выбор двух игроков должен влиять друг на друга, то получается они имеют общую функцию состояния, имеющая следующий вид:

$$|\Psi\rangle_j = \sum_{k,l=0,1}^1 a_{k,l} |kl\rangle \quad (15)$$

из этой формулы так же видно, что у игроков имеется общее ментальное состояние, что как раз более правдоподобно в реальном мире. По аналогии из формулы выше для двумерного гильбертова пространства, $a_{k,l}$ – коэффициент проекции вектора $|kl\rangle$, где $|kl\rangle$ – вектор выбора двух игроков (всего их четыре $|kl\rangle |kl\rangle |kl\rangle |kl\rangle$), а квадрат коэффициента проекции $|a_j|^2$ имеет смысл как вероятность выбора исхода игры.

3 КВАНТОВОПОДОБНАЯ МОДЕЛЬ ПРИНЯТИЯ РЕШЕНИЙ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ ВО ВЗАИМОДЕЙСТВИИ С ОКРУЖЕНИЕМ

3.1 Постановка задачи для квантовоподобной модели

Ранее был рассмотрен эксперимент с дилеммой заключенного и его сформулированный эффект. Этот эффект хорошо моделируется с помощью квантово-подобных моделей принятия решений, благодаря тому, что неопределенность описывается квантовой суперпозицией с точки зрения одного из игроков. Таким образом описание неопределенности для игрока А относительно игрока Б описывается следующим образом:

$$|\phi_B\rangle = \alpha|0_B\rangle + \beta|1_B\rangle \in \mathbb{C}^2 \quad (16)$$

где α и β - показывают степень осознания действия игрока Б, $|0_B\rangle$ - характеризует вектор предательства игроком Б, $|1_B\rangle$ - характеризует вектор сотрудничества игроком Б. \mathbb{C}^2 - двумерное комплексное пространство (в данном случае Гильбертово). Для описания ситуации с учетом принятия решения игрока А, необходимо использовать четырехмерное Гильбертово пространство $\mathbb{C}^2 \otimes \mathbb{C}^2$. В таком случае уравнение (16) имеет следующий вид, в случае предательства игрока А [21]:

$$|\Phi_{0_A}\rangle = \alpha_0|0_A0_B\rangle + \beta_0|0_A1_B\rangle \in \mathbb{C}^2 \otimes \mathbb{C}^2 \quad (17)$$

С учетом сотрудничества игрока А, вектор общего психического состояния системы двух игроков записывается следующим образом:

$$|\Psi\rangle = \sum_{x,y=0}^1 \alpha_{x,y}|x_A y_B\rangle = \alpha_0|0_A0_B\rangle + \beta_0|0_A1_B\rangle + \alpha_1|1_A0_B\rangle + \beta_1|1_A1_B\rangle \quad (18)$$

Этот подход для описания состояния двух игроков позволяет рассматривать проблему на коллективном уровне. Так например можно

применить теорию открытых квантовых систем и рассматривать игрока А как квантовую систему, а игрока Б как резервуар [22, 23]. В этом случае игрока А будем обозначать как Алиса или агент a , а игрока Б как резервуар b . В квантовой теории поля применима процедура квантования, основанная на алгебре операторов рождения и уничтожения $a, a^\dagger, b, b^\dagger$. Эти операторы позволяют рассматривать переход из одного квантового состояния в другое.

$$|1_A 0_B\rangle = a^\dagger |0_A 0_B\rangle, |0_A 1_B\rangle = a b^\dagger |1_A 0_B\rangle \quad (19)$$

Для данных операторов действуют следующие коммутационные соотношения:

$$[a, a^\dagger] = 1, [b(k), b^\dagger(k)] = \delta(k - k') \quad (20)$$

где k - размерность резервуара, $[x, y] = xy - yx$ - коммутатор. Зная коммутационное соотношение (20), можно определить числовой оператор $n_a = a^\dagger a$ для агента. Собственное значение этого оператора соответствует выбору, которым оперирует агент [23]. Для определения динамики операторов агента и резервуара используется уравнение Гейзенберга, которое имеет вид:

$$i\hbar \frac{\partial f}{\partial t} = [f, H] \quad (21)$$

где i - мнимая единица, \hbar - постоянная Планка, $\frac{\partial f}{\partial t}$ - дифференциал функции f по времени t , $[\cdot, \cdot]$ - коммутатор, H - гамильтониан. Полный гамильтониан является суммой гамильтониана агента, резервуара и их взаимодействия:

$$H = H_f + H_R + H_{int} = \hbar\omega_a a^\dagger a + \hbar \int \omega_R(k) b^\dagger(k) b(k) d^D(k) + \hbar\lambda \int g(k) (b^\dagger(k) a + a^\dagger b(k)) d^D(k) \quad (22)$$

где H_f - гамильтониан описывающий агента, H_R - гамильтониан описывающий резервуар, H_{int} - гамильтониан взаимодействия резервуара и агента, ω_a - частотная характеристика агента, $\omega_R(k)$ - частотная характеристика резервуара, $g(k)$ - параметр плотности резервуара, λ - коэффициент

взаимодействия резервуара и агента. Операторы рождения(уничтожения) агента и резервуара в представлении Гейзенберга:

$$\begin{cases} \dot{a}(t) = -i\omega_a a(t) - i\lambda \int g(k)b(k,t)d^D(k) \\ \dot{b}(k,t) = -i\omega_R(k)b(k,t) - i\lambda g(k)a(t) \end{cases} \quad (23)$$

Решение интегрально-операторного уравнения для динамики оператора $\dot{b}(k,t)$ с начальными условиями $b(k,0) = b(k)$ имеет следующий вид:

$$b(k,t) = b(k)e^{-i\omega_R(k)t} - i\lambda \int_0^t g(k)a(\tau)e^{-i\omega_R(k)(t-\tau)}d\tau \quad (24)$$

В результате полученного решения для $b(k,t)$, можно найти уравнение для $a(t)$, который имеет следующий вид:

$$\begin{aligned} \dot{a}(t) = & -i\omega_a a(t) - i\lambda \int g(k)b(k)e^{-i\omega_R(k)t}d^D(k) \\ & - \lambda^2 \iint_0^t g^2(k)a(\tau)e^{-i\omega_R(k)(t-\tau)}d\tau d^D(k) \end{aligned} \quad (25)$$

Решение этого уравнения является нетривиальной задачей, поскольку двойной интеграл в третьем слагаемом никак не сокращается. Частным решением этого уравнения является принятие постоянных значений переменных $g(k) = 1$ и $\omega_R(k) = \omega_b k$ [24]. Благодаря чему уравнение (25) имеет следующий вид:

$$\dot{a}(t) = -\left(i\omega_a + \frac{\pi\lambda^2}{\omega_b}\right)a(t) - i\lambda \int b(k)e^{-i\omega_b k t}d^D(k) \quad (26)$$

Решение такого уравнения имеет простое аналитическое решение:

$$a(t) = \left(a - i\lambda \int b(k)\eta(k,t)d^D(k)\right)e^{-\left(i\omega_a + \frac{\pi\lambda^2}{\omega_b}\right)t} \quad (27)$$

где $\eta(k,t) = \frac{1}{\rho(k)}(e^{\rho(k)t} - 1)$, $\rho(k) = i\omega - i\omega_b k + \frac{\pi\lambda^2}{\omega_b}$ [24]. Уравнение (27) подчиняется правилу коммутации. Зная свойство для операторов резервуара $\langle b^\dagger(k)b(k') \rangle = n_b \delta(k - k')$, а также свойство для динамики числового оператора

$n_a(t) = a^\dagger(t)a(t)$, который в последующем будем просто называть оператор принятия решения, можем получить решение числового оператора агента:

$$n_a(t) = n_a e^{-\frac{2\lambda\pi}{\omega_b}t} + n_b \left(1 - e^{-\frac{2\lambda\pi}{\omega_b}t}\right) \quad (28)$$

Эта модель при $t \rightarrow \infty$ стремится к фиксированному значению n_b , что как раз представлено на рисунке 3.1

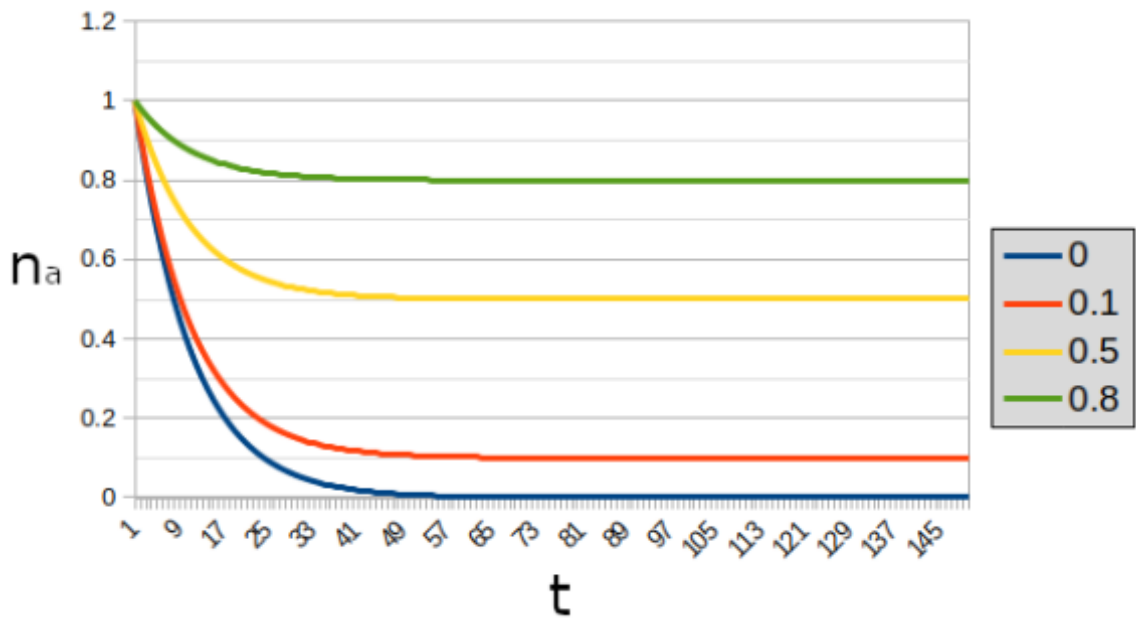


Рисунок 3.1 – Зависимость оператора принятия решения n_a агента от времени t для Марковской модели при различных параметрах информационного окружения n_b . Синяя кривая соответствует $n_b = 0$ и характеризует быстрое принятие решение агентом в отсутствии внешнего информационного поля. Красная кривая соответствует $n_b = 0.1$. Желтая кривая соответствует $n_b = 0.5$. Зеленная кривая соответствует $n_b = 0.8$ и характеризуется сильным влиянием внешнего информационного поля на агента.

Как видно из этого рисунка, характер поведения модели схож по поведению модели социально-значимых Интернет ресурсов при параметре $\rho_i = 1$. Однако, в данной модели не рассматриваются другие частные случаи для параметров $g(k)$ и $\omega_R(k)$.

3.2 Основные уравнения

В этой части рассматривается $\omega_R(k) = \omega k$, а параметр плотности $g(k)$ подчиняется следующему уравнению:

$$g^2(k) = \frac{g}{k^2 + \alpha^2}, \quad (29)$$

Подставляя уравнение (29) в уравнение (25) и приняв во внимание тот факт, что

$$\int \frac{e^{-i\omega k(t-\tau)}}{k^2 + \alpha^2} dk = \frac{\pi}{\alpha} e^{-\alpha\omega(t-\tau)} \quad (30)$$

получаем новый вид оператора $\dot{a}(t)$:

$$\dot{a}(t) = -i\omega_a a(t) - i\lambda\sqrt{g} \int \frac{b(k)e^{-i\omega kt}}{\sqrt{k^2 + \alpha^2}} dk - \frac{\lambda^2 g \pi}{\alpha} \int_0^t a(\tau) e^{-\alpha\omega(t-\tau)} d\tau \quad (31)$$

Решать данное уравнение стандартно не получится, поскольку третье слагаемое невозможно вывести из под знака интеграла. В данной работе будет использоваться два способа решения этой задачи. Первый способ будет заключаться в поиске непосредственно оператора принятия решения n_a числовым методом, не получая вид оператора агента $a^\dagger(a)$. Второй способ заключается в поиске оператора рождения(уничтожения) для агента $a^\dagger(a)$ числовым методом, а после числовым методом определить оператор принятия решения n_a .

Для первого способа, зная свойство оператора принятия решения $n_a(t) = a^\dagger(t)a(t)$, а также правило дифференцирования $(xy)' = x'y + xy'$, можно получить итерационное числовое решение этой модели:

$$n_a(t + \Delta t) = \left[i\lambda\sqrt{g} \int \frac{R_1(k, t)e^{i\omega kt} - R_2(k, t)e^{-i\omega kt}}{\sqrt{k^2 + \alpha^2}} dk - 2\frac{\lambda^2 g \pi}{\alpha} n_a(t) \right] \Delta t + n_a(t) \quad (32)$$

$$R_1(k, t + \Delta t) = \left[-i\omega_a R_1(k, t) - i\lambda \sqrt{g} \frac{n_b e^{-i\omega k t}}{\sqrt{k^2 + \alpha^2}} - \frac{\lambda^2 g \pi}{\alpha} \int_0^t R_1(k, \tau) e^{-\alpha\omega(t-\tau)} d\tau \right] \Delta t + R_1(k, t) \quad (33)$$

$$R_2(k, t + \Delta t) = \left[i\omega_a R_2(k, t) + i\lambda \sqrt{g} \frac{n_b e^{i\omega k t}}{\sqrt{k^2 + \alpha^2}} - \frac{\lambda^2 g \pi}{\alpha} \int_0^t R_2(k, \tau) e^{-\alpha\omega(t-\tau)} d\tau \right] \Delta t + R_2(k, t) \quad (34)$$

Все стадии получения этих уравнений приведены в приложении. Остается вопрос с параметрами для модели, какие они должны быть и подчиняются ли они какому-либо закону. Правило по которому параметры не могут иметь случайные величины основано на коммутационном соотношении для динамики операторов агента:

$$[\dot{a}(t), \dot{a}^\dagger(t')] = \frac{\partial}{\partial t} \frac{\partial}{\partial t'} \delta(t - t') = \dot{a}(t) \dot{a}^\dagger(t') - \dot{a}^\dagger(t') \dot{a}(t) = 0 \quad (35)$$

Это коммутационное соотношение имеет другой вид:

$$[\dot{a}(t), \dot{a}^\dagger(t')] = \omega_a^2 + \lambda^2 g \frac{\pi}{\alpha} + \frac{\lambda^4 g^2 \pi^2}{2\alpha^3 \omega} (1 - e^{-2\alpha\omega t}) = 0 \quad (36)$$

Все стадии вычислений находятся в приложении. Это уравнение можно переписать в другом виде:

$$g_{1,2} = \frac{\omega\alpha^2}{\lambda^2 \pi (1 - e^{-2\alpha\omega t})} \left(-1 \pm \sqrt{1 - 2 \frac{\omega_a^2}{\alpha\omega} (1 - e^{-2\alpha\omega t})} \right) \quad (37)$$

В дальнейшем разница между g_1 и g_2 не будет играть большой роли, поскольку это никак не будет влиять на окончательный результат модели.

Стоит отметить, что $g \in \mathbb{R}$ и $g > 0$, связано это с тем, что параметр g в уравнении (31) заключен под корнем и отрицательное значение g приведет к комплексному значению параметра принятия решения. Параметр принятия решения n_a должен быть $n_a \in \mathbb{R}$.

3.3 Методы решения для немарковского резервуара

Параметр g имеет два решения:

$$g_1 = \frac{\omega\alpha^2}{\lambda^2\pi(1 - e^{-2\alpha\omega t})} \left(-1 + \sqrt{1 - 2\frac{\omega_a^2}{\alpha\omega}(1 - e^{-2\alpha\omega t})} \right), \quad (38)$$

$$g_2 = \frac{\omega\alpha^2}{\lambda^2\pi(1 - e^{-2\alpha\omega t})} \left(-1 - \sqrt{1 - 2\frac{\omega_a^2}{\alpha\omega}(1 - e^{-2\alpha\omega t})} \right), \quad (39)$$

Ранее для этого параметра было обнаружено, что его значение не должно быть комплексным или отрицательным, поскольку этом случае для уравнения n_a решения не существует. Иначе говоря, оператор принятия решения n_a становится комплексным числом. Рассмотрим первое уравнение для параметра (38). Это уравнение имеет решение, если $\alpha, \omega, \lambda, t \neq 0$. Видно, что в начальный момент времени $t = 0$ это уравнение не применимо, однако этого не требуется по решению основного уравнения, поскольку в начальный момент времени необходимые параметры уже заданы.

$$\frac{\omega\alpha^2}{\lambda^2\pi(1 - e^{-2\alpha\omega t})}, \quad (40)$$

Первая дробь (40) имеет отрицательное значение при $\alpha < 0$. В данном уравнении ω и t не может быть отрицательным, поскольку время отсчитывается от момента наблюдения, а отрицательной частоты быть не может. Поэтому на знак влияет только два параметра. Теперь рассмотрим в уравнении (38) выражение под корнем:

$$\sqrt{1 - 2\frac{\omega_a^2}{\alpha\omega}(1 - e^{-2\alpha\omega t})} \quad (41)$$

Выражение (41) должно быть строго положительное. Теперь предположим, что $\alpha < 0, \omega > 0$. В таком случае уравнение (41) можем переписать в следующем виде:

$$\sqrt{1 + 2\frac{\omega_a^2}{|\alpha|\omega}(1 - e^{2|\alpha|\omega t})} \quad (42)$$

С увеличением времени экспонента начинает возрастать, а это значит, выражение в скобках будет отрицательное $(1 - e^{2|\alpha|\omega t}) < 0$. Раз оно отрицательное, значит второе слагаемое под корнем будет так же отрицательное в выражении (42). Поскольку все переменные в выражениях (41) и (42) являются действительными числами, то на знак под корнем может зависеть только от α . Значит, второе слагаемое в выражениях (41) и (42) всегда будет отрицательное.

В таком случае рассмотрим следующее неравенство:

$$2\frac{\omega_a^2}{\alpha\omega}(1 - e^{-2\alpha\omega t}) < 1 \quad (43)$$

Если рассмотреть предел по $t \rightarrow \infty$, то левая часть неравенства стремится к бесконечности. Изменением переменных α и ω , неравенство не соблюдается, поскольку при увеличении этих переменных, увеличивается и степень в экспоненте. Значит необходимо определить границы переменной ω_a , которые приведены в выражении (44):

$$\omega_a^2 < \frac{\alpha\omega}{2(1 - e^{2\alpha\omega t})} \quad (44)$$

Рассмотрим правую часть неравенства в пределе $t \rightarrow \infty$. Пусть в этом случае $\alpha > 0$, тогда в пределе неравенство имеет следующий вид:

$$\omega_a^2 < \frac{\alpha\omega}{2} \quad (45)$$

в случае $\alpha < 0$:

$$\omega_a^2 < \frac{\alpha\omega}{\infty} \rightarrow \omega_a^2 = 0 \quad (46)$$

причина, по которой неравенство превратилось в равенство для ω_a , заключается в неосуществимости отрицательной частоты. Случай выражения (45) легко соблюдать, однако если решать уравнение (38) с параметром $\alpha > 0$, то значение выражения (38) будет $g < 0$, что не допустимо. Значит нужно рассматривать случай для выражения (46), поскольку только в этом случае $g > 0$. Такие же значения выражения (38) при данных параметрах, справедливы

для выражения (39). Из этого заключаем вывод, что значения должны быть следующие: $\omega_a^2 = 0, \alpha < 0$.

Если внимательно рассмотреть результаты ограничений, то можно заметить отсутствие частотной характеристики агента $\omega_a^2 = 0$, чего не может происходить в рассматриваемой модели, поскольку это означает отсутствие самого агента в резервуаре. Отсутствие агента в резервуаре делает бессмысленным дальнейшее моделирование. Было выдвинуто предположение, что была совершена ошибка во время вычисления и для выявления ошибки необходимо проверить величины для каждого параметра: k - безразмерный, λ - безразмерный, $g(k)$ - [Гц], ω - [Гц], ω_a - [Гц], α - безразмерный. В результате было обнаружено, что в коммутаторе (36) не совпадают размерности при вычислении. Для чего была сделана нормировка значений по g начиная с уравнений гамильтонианов, результаты которых приведены в приложении. Однако такой способ исключения ошибки не дал результатов в уравнении (36), поэтому вопрос об ошибке в вычислениях остается открытым. Возможно ошибка случается из-за того, что дельта-функция Дирака является не дифференцируемой функцией, а значит и коммутатор (35) нельзя приравнять к нулю.

Даже несмотря на присутствие комплексной составляющей в результатах вычислений, получить результат схожий с марковскими квантово-подобными моделями оказалось невозможным для первого способа. Также, не удалось получить подобие модели социально-значимых Интернет ресурсов, но результаты и их объяснение стоит привести для понимания проблемы моделирования. Прежде всего стоит начать с тех результатов полученной модели, при котором наблюдается осцилляции в поведении модели. Разработанная модель подразумевает осцилляции при любых параметрах, и таким образом они будут присутствовать всегда, разница будет заключаться в том насколько сильно они будут влиять на конечный результат. Влияние на конечный результат может наблюдаться совершенно по разному и при

порой неожиданных результатах. У модели, разработанной данным способом, очень сложно предсказывается поведение как раз по причине дополнительной комплексной составляющей, который дает параметр g . Так например на рисунке 3.2 осцилляции очень быстро выводят агента из состояния равновесия, причем характер осцилляций даже усиливается.

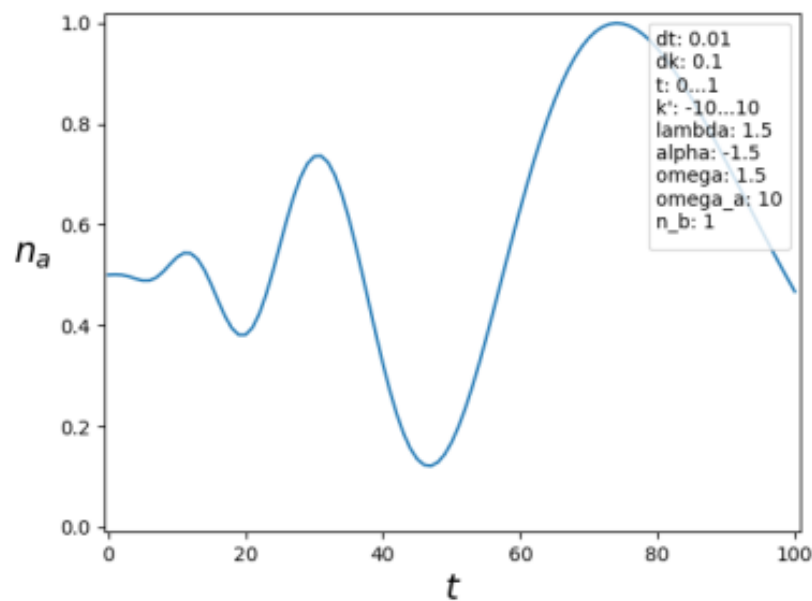


Рисунок 3.2 – Зависимость оператора принятия решения n_a агента от времени t для немарковской модели. Осциллирующее поведение процесса принятия решения у агента характеризует «метание» от одного выбора к другому. Такое поведение достигается при высоком значении частотной моды $\omega_a = 10$ у агента и не большим коэффициентом связи с информационным полем $\lambda = 1.5$

Такой результат сложно объяснить с точки зрения когнитивной психологии, поскольку экспериментов с таким эффектом найти не удалось. Результат моделирования приведенный на рисунке 3.2 можно описать как "метание" агента из одного выбора к другому, когда человеку через некоторое время кажется, что он сделал фатальную ошибку в выборе, но такое объяснение, к сожалению, не удалось привести из-за отсутствия реальных экспериментов. Помимо необъяснимости полученного результата, приведенного на рисунке 3.2, стоит отметить отсутствие комплексной плоскости на графике, поскольку присутствие комплексной составляющей

в результате моделирования дает предпосылки к ошибке вычисления, а это необходимо учитывать при моделировании. Такой же результат без комплексной плоскости необходимо отметить на следующем графике (рисунок 3.3).

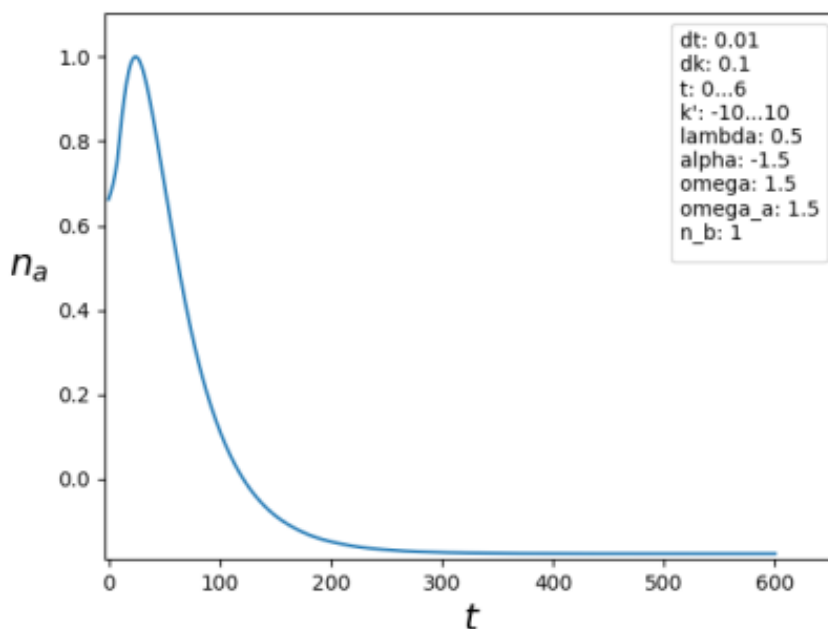


Рисунок 3.3 – Зависимость оператора принятия решения n_a агента от времени t для немарковской модели, демонстрирующее временное возбуждение (пик) с последующим равномерным затуханием и формированием принятого решения. Такое поведение достигается при малом значении частотной моды $\omega_a = 1.5$ у агента и не большим коэффициентом связи с информационным полем $\lambda = 1.5$

На рисунке 3.3 происходит некоторое временное возбуждение с последующим затуханием, поведение которого также объясняется комплексным или отрицательным параметром плотности g . В когнитивистике это поведение можно объяснить как ситуацию с отрицанием противоположной точки зрения. Когда среда начинает воздействовать на сознание противоречивой информацией, вызывая у человека когнитивный диссонанс, проявляющаяся ответной реакцией в виде пика на графике, но через некоторое время ответная реакция сознания индивида постепенно затухает, иначе говоря индивид переходит к состоянию среды.

Более правильным поведением является случай с параметром

плотности $g = 1$, по причине отсутствия комплексной составляющей в результатах модели, иначе говоря $n_a \in \mathbb{R}$. Конечно, данное действие не является правильным с точки зрения вычислений, поскольку в таком случае коммутатор в уравнении (36) не равняется нулю, а это значит что оператор рождения и уничтожения агента не коммутируют, чего не должно происходить. Поведение при $g = 1$ показано на рисунке 3.4. Как видно из рисунка 3.4

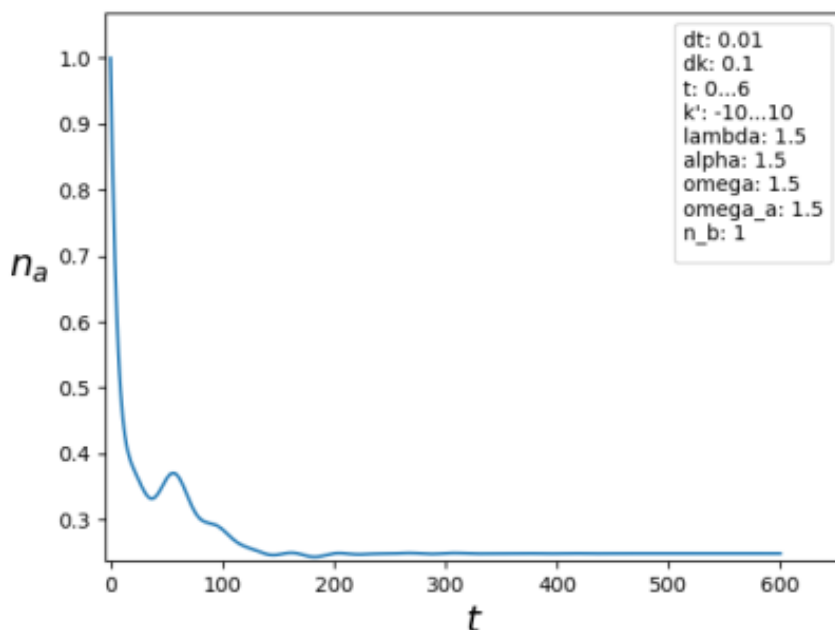


Рисунок 3.4 – Зависимость оператора принятия решения n_a агента от времени t для немарковской модели. График соответствует малому значению параметра $g = 1$, характеризующее ширину функции ядра. Поведение схоже с марковской квантово-подобной моделью.

его поведение схоже с марковской квантово-подобной моделью, про которую было сказано ранее, приведенной в уравнении (28) и на рисунке 3.1. С точки зрения квантовой когнитивистики это поведение было описано ранее в других работах [23, 24].

В заключении этой части необходимо уточнить причины проблемы присутствия комплексной составляющей в модели. Проблема заключается не только в том, что параметр плотности g вносит комплексную составляющую в оператор принятия решения $n_a \in \mathbb{C}$, но и в том, что поскольку даже

отнормировав основные уравнения по g , из-за чего в коммутаторе (36) он исчезает, отрицательными или комплексными окажутся другие параметры, так как необходимо соблюдать коммутационные соотношения. Решением этой проблемы может быть поиск аналитического способа, но в данной работе этого сделать не удалось. В таком случае необходимо найти приближенный вид аналитического решения, что может быть осуществимо.

3.4 Моделирование внешнего информационного резервуара

Для приближенного аналитического решения необходимо найти численное решение уравнения для оператора агента $a(t), a^\dagger(t)$. В основу численного решения было взято нормированное по g уравнение (47), вывод которого подробно приведен в приложении.

$$\dot{a}(t) = -i\omega'_a a(t) - i\lambda \int \frac{b(k)e^{-i\omega kt}}{\sqrt{k^2 + \alpha^2}} dk - \frac{\lambda^2 \pi}{\alpha} \int_0^t a(\tau) e^{-\alpha\omega(t-\tau)} d\tau \quad (47)$$

Как видно из уравнения (47) в нем отсутствует параметр плотности, а это значит его задавать не нужно. Применяем итерационный метод решения для уравнения (47), который имеет вид:

$$a(t + \Delta t) = -i\omega'_a a(t)\Delta t - i\lambda \int \frac{b(k)e^{-i\omega kt}}{\sqrt{k^2 + \alpha^2}} dk \Delta t - \frac{\lambda^2 \pi}{\alpha} \int_0^t a(\tau) e^{-\alpha\omega(t-\tau)} d\tau \Delta t - a(t) \quad (48)$$

Все параметры в этом уравнении задаются случайным способом в диапазоне вещественных чисел кроме одного. Этим параметром является оператор резервуара $b(k), b^\dagger(k)$, который задается в области комплексных значений. В ходе расчетов было выяснено, что большую зависимость оператор агента $a(t), a^\dagger(t)$ проявляет в зависимости от оператора резервуара $b(k), b^\dagger(k)$, а это значит необходимо продумать приближенные значения этого параметра в

итерационном методе решения. Поскольку известно, что операторы рождения и уничтожения комплексно-сопряженные между собой, то значит можно предположить отличия между оператором рождения и уничтожения только знаком комплексной составляющей. Помимо этого, параметр $b(k), b^\dagger(k)$ не просто число, а некоторая зависимость от параметра k . Такая зависимость может иметь большое количество вариантов, из которых выявить основные не получится. Моделирование оператора рождения и уничтожения резервуара $b(k), b^\dagger(k)$ нетривиальная задача, которая требует отдельного рассмотрения, но без него невозможны дальнейшие действия. В таком случае оператор рождения и уничтожения для резервуара были приняты равными константе $b(k) = b, b^\dagger(k) = b^\dagger$, при этом они должны быть комплексно-сопряженными.

После численного решения оператора рождения и уничтожения агента $a(t), a^\dagger(t)$ остается вопрос о проверке результатов. Проверка результатов может быть проведена с помощью коммутационных соотношений $[a(t), a^\dagger(t)] = 1$, которые уже были показаны в данной работе. Эти коммутационные соотношения работают только для $a(t), a^\dagger(t)$, но не для $\dot{a}(t), \dot{a}^\dagger(t)$, поэтому нужно найти хотя бы приближенное аналитическое решение. Хорошим показателем правильного моделирования является отсутствие комплексного составляющего в результате получения оператора принятия решения n_a , но для данного метода решения этого можно не делать, поскольку параметры задаются вещественными в начале моделирования. Получив итерационным методом оператор рождения и уничтожения агента $a(t), a^\dagger(t)$, можно перейти к численному результату оператора принятия решения n_a . Результаты численного решения оператора принятия решения приведены в следующей части главы.

Для данного способа решения полученные данные оказались более правильными, поскольку комплексная составляющая отсутствовала для всех результатов моделирования. Результаты моделирования хорошо показывают проявление немарковского процесса, эффекта влияния "памяти" на принятие решения, постепенный переход к равновесию в информационном резервуаре.

Разработанная модель очень просто сводится к результатам марковской квантово-подобной модели [24], если задать степень влияния осцилляций на модель достаточно слабыми, а параметр α задать гораздо больше чем в два раза частотной характеристики ω_a , что как раз приведено на рисунке 3.5.

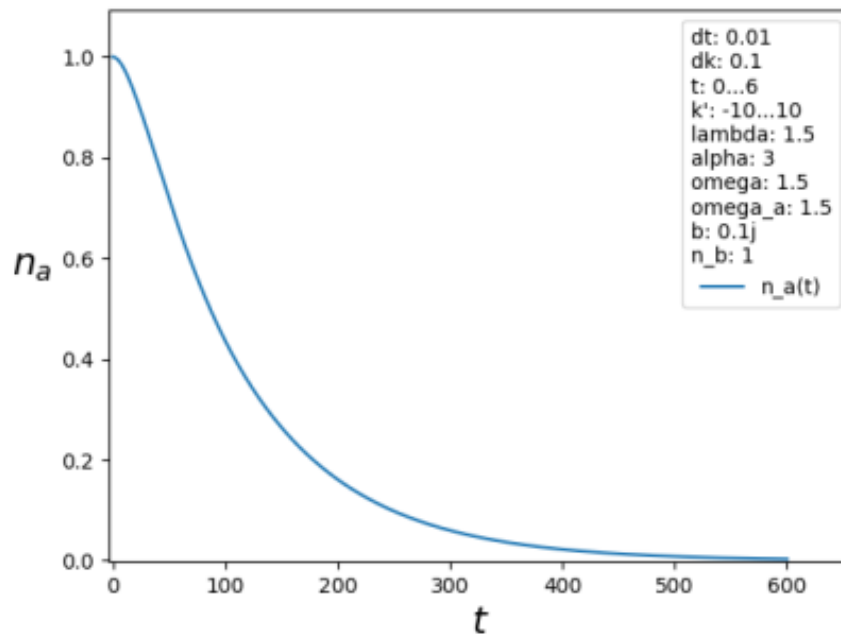


Рисунок 3.5 – Зависимость оператора принятия решения n_a агента от времени t для немарковской модели для параметра резервуара $\alpha = 3$, который характеризует марковское приближение.

Для сравнения приведенного результата на рисунке 3.5 с марковской квантово-подобной моделью принятия решения можно обратиться к рисунку 3.1. Этот результат не представляет интереса в данной работе, поскольку такие результаты обсуждались в других работах. Другой случай показывает влияние немарковского процесса на поведение агента при воздействии резервуара, который приведен на рисунке 3.6.

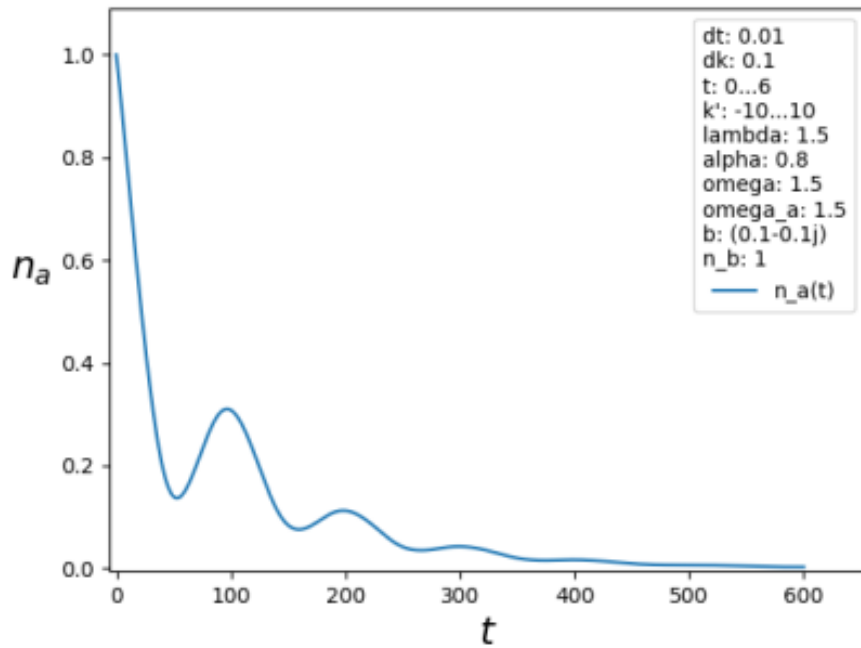


Рисунок 3.6 – Зависимость оператора принятия решения n_a агента от времени t для немарковской модели с параметром резервуара $\alpha = 0.8$, в котором появляются флуктуации в процессе принятия решения у агента.

Как видно из рисунка 3.6 агент стремится к начальному решению через некоторое время равному $t = 100, 200, 300$, происходят осцилляции с затуханием. Такое поведение, с точки зрения когнитивистики, объясняется как "когнитивный шум" и наблюдается в социальных сетях при воздействии информационной среды сообщества на участника этого сообщества. Как видно из рисунка 3.5 и рисунка 3.6 немарковский процесс может переходить в марковский, в случае малых величин оператора рождения(уничтожения) резервуара b, b^\dagger и параметра α больше частотной характеристики агента ω_a . Было обнаружено, что при уменьшении параметра α относительно частотной характеристики ω_a , увеличивается частота осцилляций модели, что в данном случае наблюдается на рисунке 3.7.

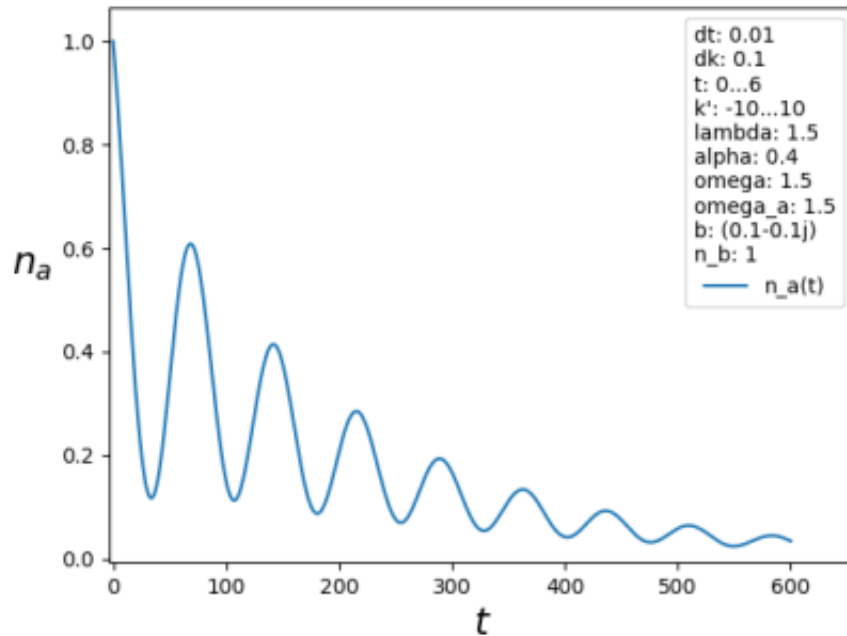


Рисунок 3.7 – Зависимость оператора принятия решения n_a агента от времени t для немарковской модели с параметром резервуара $\alpha = 0.4$, в котором проявляются очень частые флуктуации в процессе принятия решения у агента.

Необходимо отметить зависимость диапазона моделирования от величины модуля оператора рождения(уничтожения) резервуара $|b|, |b^\dagger|$. Поскольку диапазон принятия решения находится в пределах от 0 до 1, то и параметры модели должны быть подобраны таким образом, чтобы в результате соответствовать этим ограничениям. Поэтому сумма модулей оператора рождения и уничтожения резервуара не должны превышать 1, иначе говоря сумма радиусов их векторов на комплексной плоскости не должны превышать 1. На рисунке 3.7 можно заметить, что нижние полупериоды осцилляций модели достигают минимума постепенно, по закону гауссовой функции. Это поведение можно наблюдать в физике при изучении степени пространственной когерентности реальных лазерных источников в интерферометре Юнга, когда темные полосы интерференционной картины "засвечены" нескоррелированными фотонами. Такое поведение моделируется если сумма действительной части оператора рождения и уничтожения резервуара близка по сумме к 1. Более наглядное представление этого поведения представлено на рисунке 3.8.

Последний вопрос в этой части касается приближенного аналитического вида

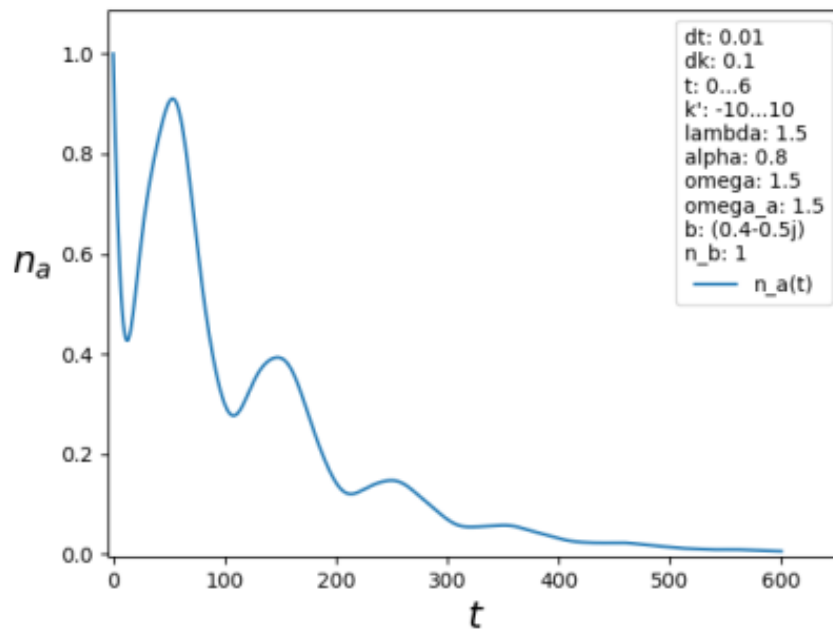


Рисунок 3.8 – Зависимость оператора принятия решения n_a агента от времени t для немарковской модели характеризующее возвышенными нижними полупериодами осцилляций, характерное в интерференционной картине.

модели принятия решения. Для решения вопроса необходимо описать основные уравнения, которые будут составлять характер модели. Первое что необходимо отметить, это присутствие осцилляций, характер которых может описать уравнение косинуса. Поскольку косинус находится в пределе от -1 до 1, то его значения нужно увеличить на единицу. Осцилляции в модели со временем затухают и их характер затухания необходимо подбирать. Для уравнения затухания было выбрано уравнение экспоненты в отрицательной степени, которая составляет произведение с уравнением косинуса. В вышеприведенных результатах поведение агента стремится к покою (к 0) и это значит присутствие экспоненты в знаменателе для всего полученного аналитического решения. Окончательный результат этой приближенной модели представлен следующим образом:

$$f(t) = \frac{\cos(tx - y) * e^{-tz} + 1}{e^t}, \quad (49)$$

где x - параметр частоты, y - параметр сдвига, z - параметр затухания

осцилляций, t - время. Для демонстрации результатов поиска приближенного решения оператора принятия решения следует привести уравнения с подобранными параметрами. Первое уравнение имеет следующий вид:

$$f(t) = \frac{\cos(10 * t/1.645 - 10) * e^{-t*0.3} + 1}{e^t}, \quad (50)$$

и его результат с приведенным немарковским поведением модели при сильных осцилляциях и их "сильных" минимумов 3.9. Второе уравнение имеет следующий

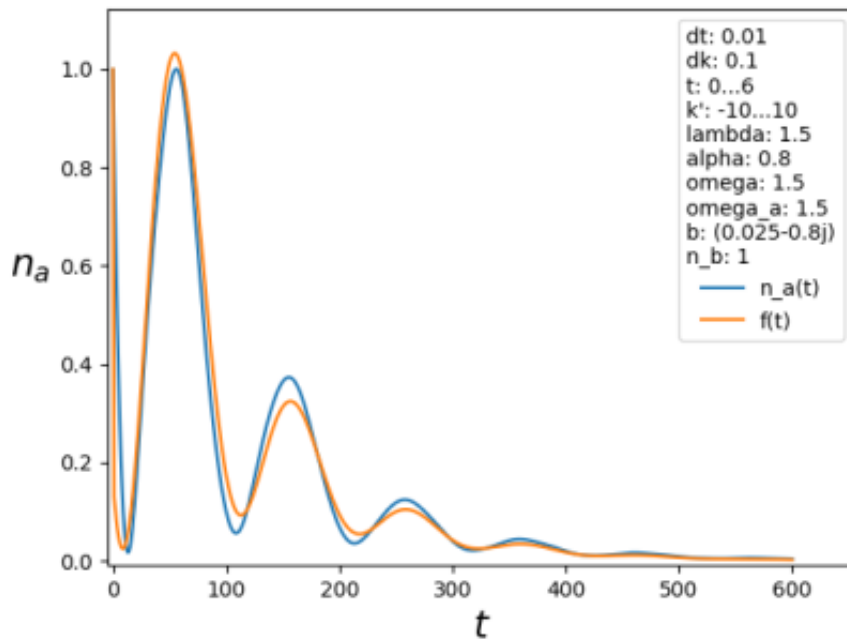


Рисунок 3.9 – Оранжевая линия – приближенное аналитическое решение, рассчитанное по формуле (50) описывающее процесс принятия решение у агента с малым значением параметра затуханий осцилляций z . Синяя кривая соответствует численному моделированию.

вид:

$$f(t) = \frac{\cos(10 * t/1.645 - 9.5) * e^{-t*0.8} + 1}{e^t}, \quad (51)$$

где его результат с вышеприведенным поведением приведен на следующем рисунке 3.10.

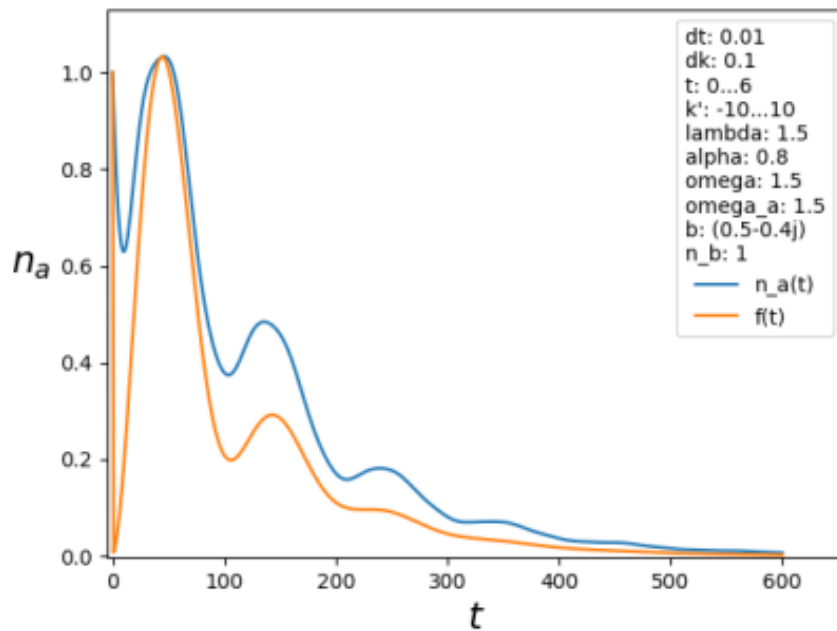


Рисунок 3.10 – Оранжевая линия – приближенное аналитическое решение, рассчитанное по формуле (51) описывающее процесс принятия решение у агента с большим значением параметра затуханий осцилляций z . Синяя кривая соответствует численному моделированию.

Коммутационное соотношение из приближенного уравнения взять не получится, поскольку отсутствует оператор рождения(уничтожения) в слагаемых. Но основная задача данной работы решена, поскольку была получена модель, описывающая эмпирические данные в социальных сетях.

Влияние оператора среды n_b оказывает на принятие решения агента и имеет линейный характер с увеличением величины этого оператора. Такой же характер имеет марковская модель принятия решения, с одним отличием, что в разработанной немарковской модели с увеличением n_b увеличивается амплитуда осцилляций. На следующем рисунке показана зависимость принятия решения агента n_a от влияния параметра оператора среды n_b 3.11.

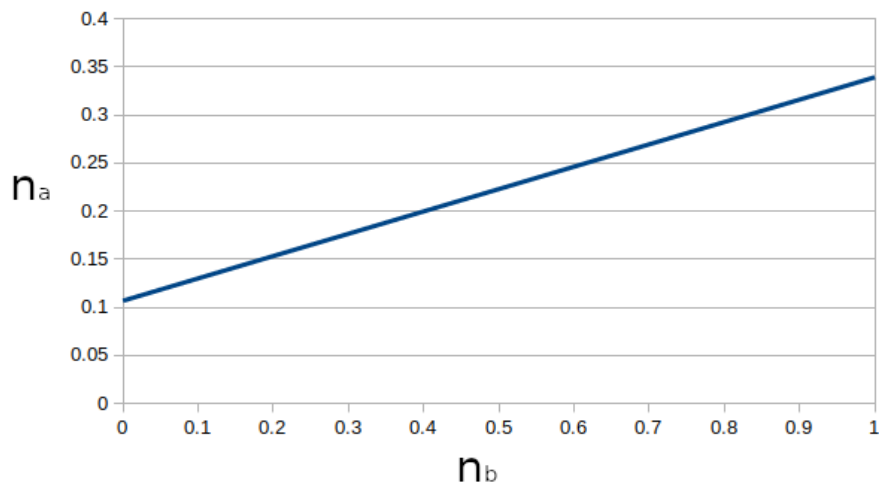


Рисунок 3.11 – Зависимость принятия решения агента n_a от величины оператора среды (информационного резервуара) n_b , при постоянном времени $t = 41$. С увеличением параметра n_b , увеличивается уровень согласованности с исходным состоянием агента.

Таким образом, разработанная модель не подменяет стандартные квантово-подобные модели принятия решения, а дополняет их, благодаря учету немарковости при разработке. Немарковость позволяет учитывать случайные процессы в мышлении человека, влияние памяти на направление мышления.

3.5 Программное моделирование

Для численного моделирования существует большое количество инструментов, из которых наиболее подходящими для данной задачи являются Wolfram Mathematica, Python, Matlab. Для численного моделирования был разработан скрипт написанный на языке программирования Python. Причины написания скрипта на этом языке заключается в доступности написанного для широкого применения. К примеру Mathematica и Matlab являются проприетарными программными продуктами, использование которых не всегда удобно для пользователей не использующих данный продукт. Основной проблемой для Python заключается в подключаемых библиотеках для поддержки определенных функций. Использование виртуальных окружений частично решает эту проблему, поскольку невозможно отследить версию

каждой библиотеки. В таком случае в открытом доступе в сети Интернет имеются сервисы для дистанционного программирования на удаленной машине с поддержкой интеллектуального подключения используемых библиотек. Существует большое сообщество программирования на Python в сети Интернет, которое может поддержать начинающих программистов в возникающих проблемах.

В поддержку преимуществ программирования на Python необходимо отметить мультипарадигменность языка, которая позволяет использовать преимущества каждой парадигмы программирования, лаконичность, которая позволяет читать исходный код скрипта более проще по сравнению с вышеприведенными языками. В разработке использовались парадигмы ООП, функционального и процедурного стиля. Для структурирования программного кода, скрипт был разделен на несколько модулей, подключаемых в основной модуль, в котором записаны начальные параметры и оформление результатов модели. Каждый модуль содержит в себе определенную структуру модели, отвечающую определенной функции в итерационном методе или функции вычисления параметров. Функции для вычисления параметров записываются в функциональном виде, поскольку использование в виде методов класса или процедур имеет ряд проблем связанных с вычислениями и структурой скрипта. Часть функций записываются в отдельном модуле, поскольку встречаются в других модулях, другая часть функций реализованы в методах класса. Итерационный метод был реализован в ООП стиле, поскольку каждая итерация имеет предыдущее состояние итерации, что значит необходимо сохранять предыдущее состояние. Предыдущее состояние системы хранится в самом классе и поэтому циклы не используются, что также улучшает читаемость кода. Благодаря ООП, код можно самодокументировать и не использовать лишние комментарии. Наследование в ООП способствует исключению лишних методов, которые могут одновременно встречаться в разных модулях. В таком случае циклы используются в случае перечисления элементов списка или матрицы. Списки

и матрицы в основном состоят из комплексных чисел, которые характеризуют некоторую структуру модели при начальных параметрах. Для преобразования комплексных чисел в область действительных и мнимых чисел используется пакет `scipy`. В случаях, где необходимо использовать мнимую и действительную часть комплексного числа для расчетов с использованием стандартных математических функций, таких как синус, косинус, квадратный корень и т.д., используется пакет `cmath`, поддерживающий вычисление комплексных чисел. вещественная и мнимая часть комплексного числа используется для получения результатов на графике. Поскольку численные методы моделирования подразумевают использование величины дискретизации, то для определения оптимальной величины выполнения программы было задано допустимое время вычисления равное не более $t = 5$ сек. и допустимая средняя арифметическая величина различия между данными с большей и меньшей частотой дискретизации по двум параметрам не более $\delta = 0.1$. Сначала приведем таблицу с временем выполнения при различных параметрах частоты дискретизации.

Таблица 4 – Таблица времени выполнения моделирования при различных значениях величины приращения и постоянных параметрах

dt	dk	время выполнения (сек.)
0.1	0.1	0.75
0.1	0.01	1.08
0.01	0.1	1.41
0.01	0.01	5.39
0.001	0.1	56.26
0.001	0.01	70.53
0.1	0.001	5.29
0.01	0.001	45.1
0.001	0.001	-

Как видно из таблицы 4 на время выполнения программы влияет в основном величина приращения времени, причем существенно много времени на расчет уходит при $dt = 0.001$, что является неприемлемым. Однако может получиться так, что точность при такой частоте дискретизации окажется на порядок высокой. Для этого необходимо найти среднее арифметическое между различными величинами приращения и сравнить их между собой.

Таблица 5 – Таблица времени выполнения моделирования при различных значениях величины приращения

	dt	dk	разница приращений						
dt			0.1	0.1	0.01	0.01	0.001	0.001	0.1
dk			0.1	0.01	0.1	0.01	0.1	0.01	0.001
	0.1	0.1	-	3.704e-5	0.053	0.053	0.056	0.056	4.074e-5
	0.1	0.01	3.704e-5	-	0.053	0.053	0.056	0.056	3.707e-6
	0.01	0.1	0.053	0.053	-	2.401e-5	0.003	0.003	0.053
	0.01	0.01	0.053	0.053	2.401e-5	-	0.003	0.003	0.053
	0.001	0.1	0.056	0.056	0.003	0.003	-	2.354e-5	0.056
	0.001	0.01	0.056	0.056	0.003	0.003	2.354e-5	-	0.057
	0.1	0.001	4.074e-5	3.707e-6	0.053	0.053	0.056	0.057	-

Как видно из таблицы 5 среднее разницы между результатами при

постоянных параметрах не выходит за пределы допустимой величины равной $\delta = 0.1$. Таким образом влияние величины приращения не сильно влияет на результат и разница точности незаметна. Если посмотреть на первую строку в таблице 5, то можно заметить разницу результатов между приращениями по dk и dt , где последний сильно влияет на точность. Так например, разница между $dt = 0.1, dk = 0.1$ и $dt = 0.1, dk = 0.01$ составляет малую величину порядка $3.704 \cdot 10^{-5}$, а разность между $dt = 0.1, dk = 0.1$ и $dt = 0.01, dk = 0.1$ уже имеет величину порядка 0.053. Разность между $dt = 0.1, dk = 0.1$ и $dt = 0.001, dk = 0.1$ составляет 0.056, что не сильно отличается от результата с $dt = 0.01, dk = 0.1$. Это более заметно если посмотреть на строку $dt = 0.001, dk = 0.1$ и столбец $dt = 0.01, dk = 0.1$, в котором записана величина увеличения точности всего лишь на 0.003. Таким образом, сравнивая результаты из таблицы 4 и 5, можно сделать вывод что наилучшими величинами приращения для расчетов являются $dt = 0.01$ и $dk = 0.1$.

Использование функционального стиля не всегда оправдано, поскольку стандартные функции могут оказаться более правильным решением в плане читаемости

кода. Таким примером является функция `reduce`, которая берет предыдущий элемент списка и последующий, операция над которыми осуществляется с помощью лямбда-функций заданными программистом, и их результат на следующей итерации становится как предыдущий элемент списка. Проще говоря, `reduce` объединяет результаты предыдущих вычислений из списка с последующими. Такой же эффект имеет стандартная функция суммирования `sum`, только в этом случае элементы списка суммируются между собой, а это значит что использование `reduce` не во всех случаях оправдано. Помимо этого ранее была использована функция интегрирования из пакета `scipy`, которая позволяла интегрировать выражение записанное лямбда-функцией без использования численного метода. Однако, использование этой функции оказалось неприемлемо, поскольку невозможно было интегрировать с учетом

предыдущего результата для итерационного метода.

Для проверки результатов выполнения программ был разработан юнит тест для функции параметра g , который проверяет проблемы с выводом значений из-за большого количества входных данных, тест для параметра принятия решения, который проверяет граничные случаи функции с разработанной моделью. Юнит тест для параметра g предназначался для определения аномалий вычисления, пример теста представлен в приложении. Для составления тестов были использованы онлайн сервисы-калькуляторы рассчитывающие численные выражения, одним из которых является Wolfram Alpha, который позволяет посмотреть введенную функцию в математическом виде. Wolfram Alpha являлся эталоном правильного решения, не позволял ошибиться человеку рассчитывая „на бумаге“. Результаты вычисления в Wolfram Alpha учитывались при тестировании и заносятся в качестве правильного решения в юнит-тест. В ходе работы этого теста, аномалии или неправильное решение уравнения g не наблюдались, что значит тест пройден. Наборы тестовых заданий подбирались таким образом, чтобы заставить уравнение g прийти либо к недопустимым значениям, либо к определенным заранее данным, либо к данным, описываемые с помощью простого прилижения. Другой подход к тестированию использовался в отношении к оператору принятия решения n_a , который можно было проверить только с помощью описания простой функции. Простые функции описывали граничное поведение, при которых n_a стремится к простой аналитической форме. Таким образом, не считая „на бумаге“ модель численно, можно было проверить, хоть и в ограниченных параметрах, параметр n_a и утверждать, что модель, по крайней мере, не выходит за пределы допустимых значений. Скрипт для тестирования n_a приведен в приложении (см. *n0_a.py*). В ходе выполнения этого теста, допустимые значения не были превышены, отсутствовали в результате комплексные числа. Таким образом, можно утверждать, что разработанная модель работает правильно.

ЗАКЛЮЧЕНИЕ

В ходе проведенной работы было изучено большинство когнитивных эффектов возникающие в условиях неопределенности. На основании изученных работ, была разработана модель принятия решения агентом с учетом его взаимодействия (обмена информацией) с внешним окружением, описываемым резервуаром. Когнитивное состояние агента в рассматриваемой модели описывается вектором в Гильбертовом пространстве, а процесс ПР в представлении вторичного квантования характеризуется операторами рождения a^\dagger и уничтожения a , которые определяют динамику ПР. Наблюдаемой величиной в рассматриваемой задаче является оператор принятия решения $n_a = a^\dagger a$, который усреднялся по когнитивным состояниям ПР и определяет основную динамику данного процесса. В зависимости от типа внешнего информационного резервуара процесс принятия решения агентом имел марковский и немарковский характер. Проведен аналитический и численный анализ модели динамического принятия решения агентом с учетом как марковости, так и не марковости резервуара, имеющего бесконечное число степеней свободы. Было показано, что влияние резервуара на агента ПР приводит к различным режимам затухания вероятности принятия решения, имеющим экспоненциальный характер в случае марковости процесса и осциллирующий характер в случае немарковости процесса. Переход от марковского к немарковскому процессу осуществляется через изменение параметров ядра взаимодействия $g(k)$ агента ПР и внешнего информационного окружения. Был проведен анализ зависимости динамики принятия решения агентом от вводимых параметров процесса, который позволил выявить, что является режимом существенно осциллирующего немарковского взаимодействия агента ПР с резервуаром. Так как немарковский процесс имеет сложную по своей структуре динамику, в работе был проведен аналитический анализ полученных результатов, который позволил получить

аналитическое решение рассматриваемой модели. Данное аналитическое решение хорошо согласуется с полученным численным моделированием модели.

СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ

ИПА - информационно-психологическая акция

ПР - принятие решения

ООП - объектно-ориентированное программирование

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Петровский, Алексей Борисович. Теория принятия решений [Текст] / Алексей Борисович Петровский // М.: Издательский центр «Академия». — 2009.
2. Демидова, ЛА. Принятие решений в условиях неопределенности [Текст] / ЛА Демидова, ВВ Кираковский, АН Пылькин // М.: Горячая линия-Телеком. — 2012.
3. Daniel, Ellsberg. Risk, ambiguity, and the Savage axioms [Текст] / Ellsberg Daniel // Quarterly Journal of Economics. — 1961. — Т. 75, № 4. — С. 643–669.
4. Tversky, Amos. The disjunction effect in choice under uncertainty [Текст] / Amos Tversky, Eldar Shafir // Psychological science. — 1992. — Т. 3, № 5. — С. 305–310.
5. Dominiak, Adam. A dynamic Ellsberg urn experiment [Текст] / Adam Dominiak, Peter Duersch, Jean-Philippe Lefort // Games and Economic Behavior. — 2012. — Т. 75, № 2. — С. 625–638.
6. Camerer, Colin. Recent developments in modeling preferences: Uncertainty and ambiguity [Текст] / Colin Camerer, Martin Weber // Journal of risk and uncertainty. — 1992. — Т. 5, № 4. — С. 325–370.
7. Shafir, Eldar. Thinking through Uncertainty: Nonconsequential Reasoning and Choice [Текст] / Eldar Shafir, Amos Tversky // preference, belief, and similarity. — 1992. — С. 703.
8. Tversky, Amos. Extensional versus intuitive reasoning: The conjunction fallacy in probability judgment. [Текст] / Amos Tversky, Daniel Kahneman // Psychological review. — 1983. — Т. 90, № 4. — С. 293.
9. Tversky, Amos. Extensional versus intuitive reasoning: The conjunction fallacy in probability judgment [Текст] / Amos Tversky, Daniel Kahneman // Preference, belief, and similarity: Selected writings by Amos Tversky. —

2004. — С. 221–56.
10. Rational cooperation in the finitely repeated prisoners' dilemma [Текст] / David M Kreps, Paul Milgrom, John Roberts, Robert Wilson // Journal of Economic theory. — 1982. — Т. 27, № 2. — С. 245–252.
 11. Shaffir, Eldar B. Typicality and reasoning fallacies [Текст] / Eldar B Shaffir, Edward E Smith, Daniel N Osherson // Memory & Cognition. — 1990. — Т. 18, № 3. — С. 229–239.
 12. Massaro, Dominic W. A pattern recognition account of decision making [Текст] / Dominic W Massaro // Memory & Cognition. — 1994. — Т. 22, № 5. — С. 616–627.
 13. Gigerenzer, Gerd. On narrow norms and vague heuristics: A reply to Kahneman and Tversky. [Текст] / Gerd Gigerenzer. — 1996.
 14. Tentori, Katya. On the determinants of the conjunction fallacy: Probability versus inductive confirmation. [Текст] / Katya Tentori, Vincenzo Crupi, Selena Russo // Journal of Experimental Psychology: General. — 2013. — Т. 142, № 1. — С. 235.
 15. von Sydow, Momme. The Bayesian logic of frequency-based conjunction fallacies [Текст] / Momme von Sydow // Journal of Mathematical Psychology. — 2011. — Т. 55, № 2. — С. 119–139.
 16. Hintikka, Jaakko. A fallacious fallacy? [Текст] / Jaakko Hintikka // Synthese. — 2004. — Т. 140, № 1/2. — С. 25–35.
 17. Пилькевич, Сергей Владимирович. Модель социально значимых Интернет-ресурсов [Текст] / Сергей Владимирович Пилькевич, Михаил Алексеевич Еремеев // Труды СПИИРАН. — 2015. — Т. 2, № 39. — С. 62–83.
 18. Colloquium: Non-Markovian dynamics in open quantum systems [Текст] / Heinz-Peter Breuer, Elsi-Mari Laine, Jyrki Piilo, Bassano Vacchini // Reviews of Modern Physics. — 2016. — Т. 88, № 2. — С. 021002.
 19. De Vega, Inés. Dynamics of non-Markovian open quantum systems [Текст] /

- Inés De Vega, Daniel Alonso // *Reviews of Modern Physics*. — 2017. — T. 89, № 1. — C. 015001.
20. Barnett, Stephen. Methods in theoretical quantum optics [Текст] / Stephen Barnett, Paul M Radmore. — [Б. м.] : Oxford University Press, 2002. — Т. 15.
 21. Quantum-like model of brain's functioning: Decision making from decoherence [Текст] / Masanari Asano, Masanori Ohya, Yoshiharu Tanaka [и др.] // *Journal of Theoretical Biology*. — 2011. — Т. 281, № 1. — С. 56–64.
 22. Bagarello, Fabio. An operator view on alliances in politics [Текст] / Fabio Bagarello // *SIAM Journal on Applied Mathematics*. — 2015. — Т. 75, № 2. — С. 564–584.
 23. Bagarello, Fabio. A quantum-like view to a generalized two players game [Текст] / Fabio Bagarello // *International Journal of Theoretical Physics*. — 2015. — Т. 54, № 10. — С. 3612–3627.
 24. Bagarello, Fabio. Quantum field inspired model of decision making: Asymptotic stabilization of belief state via interaction with surrounding mental environment [Текст] / Fabio Bagarello, Irina Basieva, Andrei Khrennikov // *Journal of Mathematical Psychology*. — 2018. — Т. 82. — С. 159–168.

А Основные аналитические методы построения модели

Рассмотрим резервуар с бесконечным числом степеней свободы k частицы которого описываются бесконечным числом оператора рождения(уничтожения) $b^\dagger(b)$. Данный резервуар определяет информационную среду, с которой взаимодействует агент ПР. ПР агента в представлении вторичного квантования характеризуется операторами рождения (уничтожения) $a^\dagger(a)$ в Гильбертовом пространстве. Эти операторы должны подчиняться следующим коммутационным соотношениям: $[\hat{a}, \hat{a}^\dagger] = 1, [\hat{b}(k), \hat{b}^\dagger(k)] = \delta(k - k')$ в то время как все остальные коммутаторы предполагаются равными нулю. Полный Гамильтониан рассматриваемой системы взаимодействия внешнего информационного поля (резервуара) и агента ПР имеет вид:

$$H_f = \hbar\omega_a a^\dagger a + \hbar\lambda \int g(k)(b^\dagger(k)a + a^\dagger b(k))d^D(k) + \hbar\lambda \int g(k)(b^\dagger(k)a + a^\dagger b(k))d^D(k), \quad (52)$$

где $\omega_a(\omega_b)$ - частотная характеристика агента(резервуара) соответственно; λ - коэффициент взаимодействия агента и резервуара; $g(k)$ - ядро, определяющее сруктуру резервуара; D - размерность пространства резервуара. В данной работе мы ограничиваемся одномерным резервуаром $D = 1$. Ядро резервуара $g(k)$ определяющее взаимодействие информационной среды и агента ПР в данной работе имеет вид:

$$g^2(k) = \frac{g}{k^2 + \alpha^2}, \quad (53)$$

Таким образом, отнормировав уравнение (52) на константу g , определяющая ширину распределения ядра мы получим:

$$H' = \omega'_a a^\dagger a + \int \omega'_R(k)b^\dagger(k)b(k)dk + \lambda \int \frac{(b^\dagger(k)a + a^\dagger b(k))}{\sqrt{k^2 + \alpha^2}}dk, \quad (54)$$

где $\omega'_a = \omega_a/g$, $\omega'_R(k) = \omega_R(k)/g$.

Решая уравнение Гейзенберга $i \frac{\partial f}{\partial t} = [f, H]$, $f \in a(a^\dagger)$ и $b(b^\dagger)$ можно получить операторные уравнения для операторов уничтожения a и b . Принимая во внимание коммутационные соотношения для операторов a и b , решение уравнения Гейзенберга для оператора b имеет вид:

$$\begin{aligned}
i\hbar \frac{\partial b(k, t)}{\partial t} &= [b(k, t), H] = \\
&= \hbar\omega'_a b(k, t)a^\dagger(t)a(t) + \hbar \int \omega'_R(k)b(k, t)b^\dagger(k, t)b(k, t)d^D(k) + \\
&+ \hbar\lambda \int \frac{(b(k, t)b^\dagger(k, t)a(t) + b(k, t)a^\dagger(t)b(k, t))}{\sqrt{k^2 + \alpha^2}} d^D(k) - \\
&- \hbar\omega'_a a^\dagger(t)a(t)b(k, t) - \hbar \int \omega'_R(k)b^\dagger(k, t)b(k, t)b(k, t)d^D(k) - \\
&- \hbar\lambda \int \frac{(b^\dagger(k, t)a(t)b(k, t) + a^\dagger(t)b(k, t)b(k, t))}{\sqrt{k^2 + \alpha^2}} d^D(k) = \\
&= xx' + yy' + zz' = \hbar\omega'_R(k)b(k, t) + \hbar\lambda \frac{a(t)}{\sqrt{k^2 + \alpha^2}}
\end{aligned}$$

где

$$\begin{aligned}
xx' &= \hbar\omega'_a b(k, t)a^\dagger(t)a(t) - \hbar\omega'_a a^\dagger(t)a(t)b(k, t) = \\
&= \hbar\omega'_a \overline{a^\dagger(t)b(k, t)a(t)} - \hbar\omega'_a \underbrace{[a^\dagger(t), b(k, t)]}_{0} a(t) - \\
&- \hbar\omega'_a a^\dagger(t) \underbrace{[a(t), b(k, t)]}_{0} - \hbar\omega'_a \overline{a^\dagger(t)b(k, t)a(t)} = \\
&= 0
\end{aligned}$$

$$\begin{aligned}
yy' &= \hbar \int \omega'_R(k)b(k, t)b^\dagger(k, t)b(k, t)d^D(k) - \\
&- \hbar \int \omega'_R(k)b^\dagger(k, t)b(k, t)b(k, t)d^D(k) = \\
&= \hbar \int \omega'_R(k) \left([b(k, t), b^\dagger(k, t)]b(k, t) + \right. \\
&+ \overline{b^\dagger(k, t)b(k, t)b(k, t)} - \overline{b^\dagger(k, t)b(k, t)b(k, t)} \left. \right) d^D(k) = \\
&= \hbar \int \omega'_R(k)\delta(k - k')b(k, t)d^D(k) = \hbar\omega'_R(k')b(k', t)
\end{aligned}$$

$$\begin{aligned}
zz' &= \hbar\lambda \int \frac{(b(k,t)b^\dagger(k,t)a(t) + b(k,t)a^\dagger(t)b(k,t))}{\sqrt{k^2 + \alpha^2}} d^D(k) - \\
&- \hbar\lambda \int \frac{(b^\dagger(k,t)a(t)b(k,t) + a^\dagger(t)b(k,t)b(k,t))}{\sqrt{k^2 + \alpha^2}} d^D(k) = \\
&= \hbar\lambda \int \frac{[b(k,t), b^\dagger(k,t)]a(t) + \overline{b^\dagger(k,t)b(k,t)a(t)}}{\sqrt{k^2 + \alpha^2}} d^D(k) - \\
&- \hbar\lambda \int \frac{\overbrace{[a^\dagger(t), b(k,t)]}^0 b(k,t) + \overline{a^\dagger(t)b(k,t)b(k,t)}}{\sqrt{k^2 + \alpha^2}} d^D(k) - \\
&- \hbar\lambda \int \frac{\overbrace{b^\dagger(k,t)[a(t), b(k,t)]}^0 + \overline{b^\dagger(k,t)b(k,t)a(t)} +}{\sqrt{k^2 + \alpha^2}} d^D(k) = \\
&+ \hbar\lambda \int \frac{\overline{a^\dagger(t)b(k,t)b(k,t)}}{\sqrt{k^2 + \alpha^2}} d^D(k) = \\
&= \hbar\lambda \int \frac{[b(k,t), b^\dagger(k,t)]a(t)}{\sqrt{k^2 + \alpha^2}} d^D(k) = \hbar\lambda \int \frac{\delta(k - k')a(t)}{\sqrt{k^2 + \alpha^2}} d^D(k) = \\
&= \hbar\lambda \frac{a(t)}{\sqrt{k^2 + \alpha^2}}
\end{aligned}$$

Решение уравнение Гейзенберга для оператора a :

$$\begin{aligned}
i\hbar \frac{\partial a(t)}{\partial t} &= [a(t), H] = \\
&= \hbar\omega'_a a(t)a^\dagger(t)a(t) + \hbar \int \omega'_R(k)a(t)b^\dagger(k,t)b(k,t)d^D(k) + \\
&+ \hbar\lambda \int \frac{(a(t)b^\dagger(k,t)a(t) + a(t)a^\dagger(t)b(k,t))}{\sqrt{k^2 + \alpha^2}} d^D(k) - \\
&- \hbar\omega'_a a^\dagger(t)a(t)a(t) - \hbar \int \omega'_R(k)b^\dagger(k,t)b(k,t)a(t)d^D(k) - \\
&- \hbar\lambda \int \frac{(b^\dagger(k,t)a(t)a(t) + a^\dagger(t)b(k,t)a(t))}{\sqrt{k^2 + \alpha^2}} d^D(k) = \\
&= xx' + yy' + zz' = \hbar\omega'_a a(t) + \hbar\lambda \int \frac{b(k,t)}{\sqrt{k^2 + \alpha^2}} d^D(k)
\end{aligned}$$

где

$$\begin{aligned}
xx' &= \hbar \omega'_a a(t) a^\dagger(t) a(t) - \hbar \omega'_a a^\dagger(t) a(t) a(t) = \\
&= \hbar \omega'_a \underbrace{[a(t), a^\dagger(t)]}_1 a(t) + \cancel{\hbar \omega'_a a^\dagger(t) a(t) \overline{a(t)}} - \cancel{\hbar \omega'_a a^\dagger(t) a(t) \overline{a(t)}} = \\
&= \hbar \omega'_a a(t) \\
yy' &= \hbar \int \omega'_R(k) \overbrace{a(t) b^\dagger(k, t) b(k, t)}^{=b^\dagger(k, t) a(t) b(k, t)} d^D(k) \\
&\quad - \hbar \int \omega'_R(k) \overbrace{b^\dagger(k, t) b(k, t) a(t)}^{=b^\dagger(k, t) a(t) b(k, t)} d^D(k) = 0 \\
zz' &= \hbar \lambda \int \frac{(a(t) b^\dagger(k, t) a(t) + a(t) a^\dagger(t) b(k, t))}{\sqrt{k^2 + \alpha^2}} d^D(k) - \\
&\quad - \hbar \lambda \int \frac{(b^\dagger(k, t) a(t) a(t) + a^\dagger(t) b(k, t) a(t))}{\sqrt{k^2 + \alpha^2}} d^D(k) = \\
zz' &= \hbar \lambda \int \frac{\overbrace{[a(t), b^\dagger(k, t)]}^0 a(t) + \cancel{b^\dagger(k, t) a(t) \overline{a(t)}}}{\sqrt{k^2 + \alpha^2}} d^D(k) + \\
&\quad + \hbar \lambda \int \frac{[a(t), a^\dagger(t)] b(k, t) + \cancel{a^\dagger(t) a(t) \overline{b(k, t)}}}{\sqrt{k^2 + \alpha^2}} d^D(k) - \\
&\quad - \hbar \lambda \int \frac{\cancel{b^\dagger(k, t) a(t) \overline{a(t)}} - a^\dagger(t) \overbrace{[a(t), b(k, t)]}^0 + \cancel{a^\dagger(t) a(t) \overline{b(k, t)}}}{\sqrt{k^2 + \alpha^2}} d^D(k) = \\
&= \hbar \lambda \int \frac{[a(t), a^\dagger(t)] b(k, t)}{\sqrt{k^2 + \alpha^2}} d^D(k) = \hbar \lambda \int \frac{b(k, t)}{\sqrt{k^2 + \alpha^2}} d^D(k)
\end{aligned}$$

В результате, мы имеем следующий вид операторных уравнений для операторов уничтожения a и b

$$\left\{ \begin{aligned} \dot{a}(t) &= -i \omega'_a a(t) - i \lambda \int \frac{b(k, t)}{\sqrt{k^2 + \alpha^2}} d^D(k) \\ \dot{b}(k, t) &= -i \omega'_R(k) b(k, t) - i \lambda \frac{a(t)}{\sqrt{k^2 + \alpha^2}} \end{aligned} \right. \quad (62)$$

Решение системы ищется адиабатически исключая резервуар. То есть, в начале решается второе уравнение системы (62) на $b(k, t)$, а затем его решение подставляется в первое уравнение на $\dot{a}(t)$. Физически это означает, что сперва формируется резервуар, а лишь затем он формирует свойства агента. Уравнение на $b(k, t)$ является неоднородным дифференциальным уравнением, решение которого имеет вид суммы соответствующего линейного однородного дифференциального уравнения (ЛОДУ) и частного решения неоднородного дифференциального уравнения (ЛНДУ). Решение ЛОДУ имеет вид:

$$\begin{aligned}\frac{\partial b(k, t)}{\partial t} &= -i\omega'_R(k)b(k, t); \\ b(k, t) &= Ce^{-i\omega'_R(k)t}\end{aligned}\tag{63}$$

В общем случае $C = C(t)$ и для определения данной функции необходимо подставить (63) в соответствующее ЛНДУ, таким образом:

$$\begin{aligned}(C(t)e^{-i\omega'_R(k)t})'_t &= -i\omega'_R(k)C(t)e^{-i\omega'_R(k)t} - i\lambda \frac{a(t)}{\sqrt{k^2 + \alpha^2}} \\ C(t) &= -i\lambda \int \frac{a(t)e^{i\omega'_R(k)t}}{\sqrt{k^2 + \alpha^2}} dt + C_1\end{aligned}\tag{65}$$

Объединяя (63) и (65) получим решение оператора $b(k, t)$ в следующем виде:

$$b(k, t) = b(k)e^{-i\omega'_R(k)t} - i\lambda \int_0^t \frac{a(\tau)e^{-i\omega'_R(k)(t-\tau)}}{\sqrt{k^2 + \alpha^2}} d\tau$$

Данное решение легко проверить, подставив его в (62).

$$\begin{aligned}
& \left(b(k) e^{-i\omega'_R(k)t} - i\lambda \int_0^t \frac{a(\tau) e^{-i\omega'_R(k)(t-\tau)}}{\sqrt{k^2 + \alpha^2}} d\tau \right)'_t = \\
& = b(k) \left(e^{-i\omega'_R(k)t} \right)'_t - i\lambda \left(\int_0^t \frac{a(\tau) e^{-i\omega'_R(k)(t-\tau)}}{\sqrt{k^2 + \alpha^2}} d\tau \right)'_t = \\
& = -i\omega'_R(k) \underbrace{b(k) e^{-i\omega'_R(k)t}}_{b(k,0)=b(k,t)} - i\lambda \frac{a(t) e^{-i\omega'_R(k)(t-t)}}{\sqrt{k^2 + \alpha^2}} = \\
& = -i\omega'_R(k) b(k, t) - i\lambda \frac{a(t)}{\sqrt{k^2 + \alpha^2}}
\end{aligned}$$

Мы принимаем $\omega'_R(k) = \omega k$ опираясь на известное волновое соотношение в оптике. С учетом того, что аналитическое решение интеграла имеет вид:

$$\int \frac{e^{-i\omega k(t-\tau)}}{k^2 + \alpha^2} dk = \frac{\pi}{\alpha} e^{-\alpha\omega(t-\tau)},$$

то в адиабатическом приближении решение системы (62) имеет вид:

$$\dot{a}(t) = -i\omega'_a a(t) - i\lambda \int \frac{b(k) e^{-i\omega k t}}{\sqrt{k^2 + \alpha^2}} dk - \frac{\lambda^2 \pi}{\alpha} \int_0^t a(\tau) e^{-\alpha\omega(t-\tau)} d\tau \quad (69)$$

Решить данное интегрально-операторное уравнение явно не получится из-за специфики данного уравнения, поэтому дальнейший анализ модели осуществлялся численно. В работе использовался итерационный метод для решения данного уравнения и поиска наблюдаемой величина $n_a = \langle a^\dagger(t) a(t) \rangle$, где усреднение проводится по когнитивным состояниям ПР. Для численного решения переведем интегральное уравнение в дифференциальное, таким образом:

$$\frac{\partial n_a(t)}{\partial t} = \frac{\partial (a^\dagger(t) a(t))}{\partial t} = \frac{\partial a^\dagger(t)}{\partial t} a(t) + a^\dagger(t) \frac{\partial a(t)}{\partial t} = \dot{a}^\dagger(t) a(t) + a^\dagger(t) \dot{a}(t). \quad (70)$$

Используя уравнение (69) мы можем получить следующие соотношения:

$$\dot{a}^\dagger(t)a(t) = i\omega'_a a^\dagger(t)a(t) + i\lambda \int \frac{b^\dagger(k)a(t)e^{i\omega kt}}{\sqrt{k^2 + \alpha^2}} dk - \frac{\lambda^2 \pi}{\alpha} \int_0^t a^\dagger(\tau)a(t)e^{-\alpha\omega(t-\tau)} d\tau$$

$$a^\dagger(t)\dot{a}(t) = -i\omega'_a a^\dagger(t)a(t) - i\lambda \int \frac{a^\dagger(t)b(k)e^{-i\omega kt}}{\sqrt{k^2 + \alpha^2}} dk - \frac{\lambda^2 \pi}{\alpha} \int_0^t a^\dagger(t)a(\tau)e^{-\alpha\omega(t-\tau)} d\tau$$

Подставляя данные выражения в исходное соотношение (70) можем получить дифференциальное уравнение наблюдаемой величины $n_a(t)$ в следующем виде:

$$\begin{aligned} \frac{\partial n_a(t)}{\partial t} = i\lambda \int \frac{(b^\dagger(k)a(t)e^{i\omega kt} - a^\dagger(t)b(k)e^{-i\omega kt})}{\sqrt{k^2 + \alpha^2}} dk - \\ - \frac{\lambda^2 \pi}{\alpha} \int_0^t e^{-\alpha\omega(t-\tau)} \overbrace{(a^\dagger(\tau)a(t) + a^\dagger(t)a(\tau))}^{n_a(t)\delta(t-\tau) + n_a(t)\delta(t-\tau)} d\tau \quad (73) \end{aligned}$$

в простом виде выражение (70) можно переписать следующим образом:

$$\frac{\partial n_a(t)}{\partial t} = i\lambda \int \frac{(b^\dagger(k)a(t)e^{i\omega kt} - a^\dagger(t)b(k)e^{-i\omega kt})}{\sqrt{k^2 + \alpha^2}} dk - 2\frac{\lambda^2 \pi}{\alpha} n_a(t) \quad (74)$$

Данное выражение позволяет отследить как динамику процесса принятия решение у агента. Третье слагаемое данного выражение имеет смысл ассимптотической стабилизации, при которой $n_a(t)$ уменьшается со временем. В этом случае исключая второе слагаемое мы получим результат описанный в работе $n_a(t) = n_a e^{-\frac{2\lambda^2 \pi}{\alpha} t}$ описывающий изначальню пустой резервуар. Второе слагаемое в выражении (74) описывает немарковское воздействие резервуара на агента ПР. Для численного анализа данного слагаемого необходимо определить $(b^\dagger(k)a(t))$ и $a^\dagger(t)b(k)$. Для их нахождения используем (69), в этом случае

выражение имеет вид:

$$\begin{aligned} \frac{\partial(b^\dagger(k)a(t))}{\partial t} &= b^\dagger(k)\dot{a}(t) = -i\omega'_a b^\dagger(k)a(t) - \\ &- i\lambda \int \frac{b^\dagger(k)b(k)e^{-i\omega kt}}{\sqrt{k^2 + \alpha^2}} dk - \frac{\lambda^2 \pi}{\alpha} \int_0^t b^\dagger(k)a(\tau)e^{-\alpha\omega(t-\tau)} d\tau \end{aligned} \quad (75)$$

$$\begin{aligned} \frac{\partial(a^\dagger(t)b(k))}{\partial t} &= \dot{a}^\dagger(t)b(k) = i\omega'_a a^\dagger(t)b(k) + \\ &+ i\lambda \int \frac{b^\dagger(k)b(k)e^{i\omega kt}}{\sqrt{k^2 + \alpha^2}} dk - \frac{\lambda^2 \pi}{\alpha} \int_0^t a^\dagger(\tau)b(k)e^{-\alpha\omega(t-\tau)} d\tau \end{aligned} \quad (76)$$

принимая во внимание коммутационные соотношения, получим следующие интегрально-дифференциальные уравнения:

$$b^\dagger(k)\dot{a}(t) = -i\omega'_a b^\dagger(k)a(t) - i\lambda \frac{n_b e^{-i\omega kt}}{\sqrt{k^2 + \alpha^2}} - \frac{\lambda^2 \pi}{\alpha} \int_0^t b^\dagger(k)a(\tau)e^{-\alpha\omega(t-\tau)} d\tau \quad (77)$$

$$\dot{a}^\dagger(t)b(k) = i\omega'_a a^\dagger(t)b(k) + i\lambda \frac{n_b e^{i\omega kt}}{\sqrt{k^2 + \alpha^2}} - \frac{\lambda^2 \pi}{\alpha} \int_0^t a^\dagger(\tau)b(k)e^{-\alpha\omega(t-\tau)} d\tau \quad (78)$$

Используя конечно-разностную схему второго порядка $f'(x) = \lim_{\Delta x \rightarrow 0} \frac{f(x+\Delta x) - f(x)}{\Delta x}$ можно записать полученные выражения (74), (77) и (78) в следующем виде:

$$\begin{aligned} n_a(t + \Delta t) &= \left[i\lambda \int \frac{R_1(k, t)e^{i\omega kt} - R_2(k, t)e^{-i\omega kt}}{\sqrt{k^2 + \alpha^2}} dk - \right. \\ &\quad \left. - 2\frac{\lambda^2 \pi}{\alpha} n_a(t) \right] \Delta t + n_a(t) \end{aligned} \quad (79)$$

$$\begin{aligned} R_1(k, t + \Delta t) &= \left[-i\omega'_a R_1(k, t) - i\lambda \frac{n_b e^{-i\omega kt}}{\sqrt{k^2 + \alpha^2}} - \right. \\ &\quad \left. - \frac{\lambda^2 \pi}{\alpha} \int_0^t R_1(k, \tau)e^{-\alpha\omega(t-\tau)} d\tau \right] \Delta t + R_1(k, t) \end{aligned} \quad (80)$$

$$R_2(k, t + \Delta t) = \left[i\omega'_a R_2(k, t) + i\lambda \frac{n_b e^{i\omega k t}}{\sqrt{k^2 + \alpha^2}} - \frac{\lambda^2 \pi}{\alpha} \int_0^t R_2(k, \tau) e^{-\alpha\omega(t-\tau)} d\tau \right] \Delta t + R_2(k, t) \quad (81)$$

где $R_1(k, t) = b^\dagger(k)a(t)$, $R_2(k, t) = a^\dagger(t)b(k)$. Решение поставленной задачи сводится к решению трех уравнений описанных выше. Их решение позволит отследить динамику ПР у агента и влияние структуры резервуара на принятие решения.

Б Исходный код

Листинг Б.1 – n.py

```
1 from numpy import arange
2 from cmath import exp, sqrt, pi
3 from functools import *
4 import g
5 import math
6 import R
7
8 i = 1j
9
10 class Alice:
11     def __init__(self, arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, n_b):
12         self.n = [1]
13         self.arr_t = arr_t
14         self.arr_ks = arr_ks
15         self.r1 = R.Main(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, n_b, True)
16         self.r2 = R.Main(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, n_b, False)
17         self.g = g.calc(lamda, alpha, omega, omega_a)
18         self.one_calc = self.part_one(self.r1, self.r2, lamda, self.g, alpha, omega, dk, dt)
19         self.two_calc = self.part_two(lamda, self.g, alpha, dt)
20         [ self.calc(it,t) for (it,t) in enumerate(arr_t)]
21
22     def getAll(self):
23         return self.n
24
25     def part_one(self, r1, r2, lamda, g, alpha, omega, dk, dt):
26         return lambda iks, it, ks, t: i*lamda*sqrt(g(t))* \
27             (exp(i*omega*ks*t)*r1.get(iks,it)-exp(-i*omega*ks*t)*r2.get(iks,it))/sqrt(ks**2+alpha**2)*dk*dt
28
29     def part_two(self, lamda, g, alpha, dt):
30         return lambda n_a, t: 2*lamda**2*g(t)*pi/alpha*n_a*dt
31
32     def calc(self, it, t):
33         self.n = self.n + [sum([ self.one_calc(iks,it,ks,t) for (iks,ks) in enumerate(self.arr_ks)])-self.two_calc(self.n[-1],t) + self
34             .n[-1]]
```

Листинг Б.2 – g.py

```
1 import math
2 from cmath import exp, sqrt, pi
3
4
5 def calc(my_lambda, alpha, omega, omega_a):
6     # return lambda t: omega**2*alpha**3*(-1+sqrt(1-2*omega_a**2*(1-math.exp(-2*alpha*omega**2*t)))/(alpha**2*omega**2)) /\
7     #     (my_lambda**2*math.pi*(1-math.exp(-2*alpha*omega**2*t)))
8     # return lambda t: omega*alpha**2*(-1+sqrt(1-2*omega_a**2*(1-math.exp(-2*alpha*omega*t)))/(alpha*omega)) /\
9     #     (my_lambda**2*math.pi*(1-math.exp(-2*alpha*omega*t)))
10    return lambda t: 1+t*0
11    # return lambda t: omega**2*alpha**2*(-1+sqrt(1-2*(omega_a**2-1)*(1-math.exp(-2*alpha*omega**2*t)))/(alpha*omega**2)) /\
12    #     (my_lambda**2*math.pi*(1-math.exp(-2*alpha*omega**2*t)))
```

Листинг Б.3 – R.py

```
1 from numpy import arange
2 from cmath import exp, sqrt, pi
3 from functools import *
4 import g
5 import math
6
7 i = 1j
8
9 class Main:
```

```

10 def __init__(self, arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, n_b, bsign:bool):
11     self.R = [ [1] for x in arr_ks ]
12     self.arr_t = arr_t
13     self.arr_ks = arr_ks
14     sign = -1 if bsign == True else 1
15     self.g = g.calc(lamda, alpha, omega, omega_a)
16     self.one_calc = self.part_one(sign, omega_a, dt)
17     self.two_calc = self.part_two(sign, lamda, self.g, omega, alpha, n_b, dt)
18     self.three_calc = self.part_three(lamda, self.g, omega, alpha, dt)
19     [ [ self.calc(iks,it,ks,t) for (it,t) in enumerate(arr_t) ] for (iks,ks) in enumerate(arr_ks) ]
20
21 def get(self, iks, it):
22     return self.R[iks][it]
23
24 def getAll(self):
25     return self.R
26
27 def part_one(self, sign, omega_a, dt):
28     return 1/(1-sign*i*omega_a*dt)
29
30 def part_two(self, sign, lamda, g, omega, alpha, n_b, dt):
31     return lambda t, ks: sign*i*lamda*sqrt(g(t))*n_b*dt*exp(sign*i*omega*ks*t)/sqrt(ks**2+alpha**2)
32
33 def part_three(self, lamda, g, omega, alpha, dt):
34     return lambda Riks, t: lamda**2*g(t)*pi*dt/alpha \
35         *sum([ Rtau*math.exp(-alpha*omega*(t-tau))*dt for (Rtau,tau) in zip(Riks,arange(0,t,dt)) ])
36
37 def calc(self, iks, it, ks, t):
38     self.R[iks] = self.R[iks] + [ self.one_calc * (self.two_calc(t,ks) + self.R[iks][it-1] - self.three_calc(self.R[iks], t)) ]

```

Листинг Б.4 – a.py

```

1 from numpy import arange
2 from cmath import exp, sqrt, pi
3 from functools import *
4 import g
5 import math
6
7 i = 1j
8
9 class Main:
10     def __init__(self, arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, a_0, b, bsign:bool):
11         self.a = [a_0]
12         self.arr_t = arr_t
13         self.dt = dt
14         self.arr_ks = arr_ks
15         self.dk = dk
16         sign = -1 if bsign == True else 1
17         self.one_calc = self.part_one(sign, omega_a)
18         self.two_calc = self.part_two(sign, dk, lamda, alpha, omega, b)
19         self.three_calc = self.part_three(dt, lamda, alpha, omega)
20         [ self.calc(it,t) for (it,t) in enumerate(arr_t) ]
21
22     def getAll(self):
23         return self.a
24
25     def part_one(self, sign, omega_a):
26         return lambda a: sign*i*omega_a*a[-1]
27
28     def part_two(self, sign, dk, lamda, alpha, omega, b):
29         return lambda t: sum([ sign*i*lamda*b*exp(sign*i*omega*kt)/sqrt(k**2+alpha**2)*dk for k in self.arr_ks ])
30
31     def part_three(self, dt, lamda, alpha, omega):
32         return lambda a, t: -lamda**2*pi/alpha*sum([ atau*exp(-alpha*omega*(t-tau))*dt for (atau,tau) in zip(a, self.arr_t) ])
33
34     def calc(self, it, t):
35         self.a = self.a + [ self.dt*(self.one_calc(self.a)+self.two_calc(t)+self.three_calc(self.a,t)) + self.a[-1]]

```

Листинг Б.5 – test.py

```
1 import unittest
2 import math
3 import g
4
5 ## g.calc(lambda, alpha, omega, omega_a)
6
7 class TestG(unittest.TestCase):
8
9     def test_omega_a_null(self):
10         G = g.calc(1, 1, 1, 0)
11         self.assertEqual(G(1), 0)
12
13     def test_omega_null(self):
14         G = g.calc(1, 1, 0, 1)
15         with self.assertRaises(ZeroDivisionError):
16             G(1)
17
18     def test_alpha_null(self):
19         G = g.calc(1, 0, 1, 1)
20         with self.assertRaises(ZeroDivisionError):
21             G(1)
22
23     def test_time_null(self):
24         G = g.calc(1, 1, 1, 1)
25         with self.assertRaises(ZeroDivisionError):
26             G(0)
27
28     def test_lambda_null(self):
29         G = g.calc(0, 1, 1, 1)
30         with self.assertRaises(ZeroDivisionError):
31             G(1)
32
33     def test_one(self):
34         G = g.calc(0.5, 1.5, 1.5, 1.5)
35         self.assertAlmostEqual(G(5), -6.445775195221776+3.7214700440970883j, 5)
36         G = g.calc(0.5, -1.5, 10, 1.5)
37         self.assertAlmostEqual(G(1), 1.4748507241626008e-128-3.560253963687075e-64j, 5)
38         G = g.calc(0.5, 1.5, 10, 1.5)
39         self.assertAlmostEqual(G(1), -4.329905031706483+0j, 5)
40         G = g.calc(0.5, -1.5, 1.5, 0.00005)
41         self.assertAlmostEqual(G(1), 4.774649801380661e-09-0j, 5)
42
43 if __name__ == '__main__':
44     unittest.main()
```

Листинг Б.6 – main.py

```
1 from cmath import exp, sqrt, pi, sin, cos
2 from numpy import arange
3 from functools import *
4 import matplotlib.pyplot as plt
5 import scipy
6 import n
7 import D
8 import n0_a
9 import g
10 import R
11 import a
12 import csv
13
14
15 t = 6
16 dt = 0.01
17 ks = 10
18 dk = 0.001
19
```

```

20 lamda = 1.5
21 alpha = 0.8
22 omega = 1.5
23 omega_a = 1.5
24 n_b = 1
25 b_a = 0.5-0.4j
26 b_c = 0.5+0.4j
27
28 arr_t = arange(0+dt,t+dt,dt)
29 arr_ks = arange(-ks,ks+dk,dk)
30 #r1 = R.Main(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, n_b, True)
31 #r2 = R.Main(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, n_b, False)
32 #n_a = n.Alice(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, n_b)
33 #a_annihilator = a.Main(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, 1, 0.1j, True)
34 #a_creator = a.Main(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, 1, 0.1j, False)
35 #b_a = 0.1j
36 a_annihilator = a.Main(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, 1, 0.1-0.1j, True)
37 a_creator = a.Main(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, 1, 0.1+0.1j, False)
38 b_a = 0.1-0.1j
39 #a_annihilator = a.Main(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, 1, 0.4-0.5j, True)
40 #a_creator = a.Main(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, 1, 0.4+0.5j, False)
41 #b_a = 0.4-0.5j
42 #a_annihilator = a.Main(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, 1, 0.025-0.8j, True)
43 #a_creator = a.Main(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, 1, 0.025+0.8j, False)
44 #b_a = 0.025-0.8j
45 #a_annihilator = a.Main(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, 1, b_a, True)
46 #a_creator = a.Main(arr_t, dt, arr_ks, dk, lamda, alpha, omega, omega_a, 1, b_c, False)
47 #d = D.calc(dt, _k, lamda, G, alpha, omega, omega_a)
48 #n0 = n0_a.calc(dt, alpha, omega, omega_a)
49
50 text = "dt: " + str(dt) + "\n" + \
51        "dk: " + str(dk) + "\n" + \
52        "t: 0..." + str(t) + "\n" + \
53        "k': " + str(-ks) + "... " + str(ks) + "\n" + \
54        "lambda: " + str(lamda) + "\n" + \
55        "alpha: " + str(alpha) + "\n" + \
56        "omega: " + str(omega) + "\n" + \
57        "omega_a: " + str(omega_a) + "\n" + \
58        "b: " + str(b_a) + "\n" + \
59        "n_b: " + str(n_b)
60
61
62 #n_real = [ scipy.real(elem) for elem in n_a.getAll()]
63 #n_max = max(n_real)
64 #n_real = [ (elem/n_max+1)/2 for elem in n_real]
65 #x, y = int(1/dt)/2, (max(n_real)-min(n_real))/2 + min(n_real)
66 #plt.plot(n_real)
67 #with open( str(n_b) + ".csv", "a", newline="" ) as file:
68 #    writer = csv.writer(file)
69 #    writer.writerow(n_real)
70 #plt.text(x, y, text)
71 #a_real = [ scipy.real(elem) for elem in a_annihilator.getAll()]
72 #a_imag = [ scipy.imag(elem) for elem in a_annihilator.getAll()]
73 #plt.plot(a_real, label="a(t)")
74 #plt.plot(a_imag, label="i*a(t)")
75 #plt.plot(a_creator, getAll(), label="a(t)+")
76 plt.plot([ ac*aa for (ac,aa) in zip(a_creator.getAll(), a_annihilator.getAll()) ], label="n_a(t)")
77 #plt.yscale('log')
78 #plt.show()
79
80 #print(r1.getAll())
81 #print(r2.getAll())
82 #print(n_a.getAll())
83 #print(a_annihilator.getAll())
84 #print(a_creator.getAll())
85 #G = g.calc(lamda, alpha, omega, omega_a)
86 #print([ G(t) for t in arr_t])
87
88

```



```

89 ## check integrate in 4) [da(t),da+(t)]
90 #delta_fun = lambda tau,taus: 1 if (tau-taus)==0 else 0
91 #delta_fun = lambda tau,taus: sin(9999*pi*(tau-taus))/(pi*(tau-taus))
92 #result = reduce(lambda prev,next: prev+next, [exp(-alpha*omega*(2*t-tau-taus))*delta_fun(tau,taus)*dt for tau in arr_t for taus in
    arr_t])
93 #print(result)
94 #print((1-exp(-2*alpha*omega*t))/(2*alpha*omega))
95
96 ## function Pilkevich
97 l = 0.3
98 rho = 10
99 beta = [1]
100 beta = beta + [ beta[-1]*(cos(rho*sqrt(t-0)))*exp(-l*(t-0)) for t in arr_t]
101 beta_t = [1]
102 #beta_t = beta_t + [ (beta_t[-1]*(cos(rho*t/1.645-10)*exp(-l*t))+1)/exp(t) for t in arr_t]
103 #beta_t = beta_t + [ (cos(rho*t/1.645-9.5)*exp(-t*0.8)+1)/exp(t) for t in arr_t]
104 #beta_ar = beta_t + [ cos(rho*t/2.65)/(t*1.5+1) for t in arr_t]
105 #beta_ai = beta_t + [ -sin(rho*t/2.65)*exp(-t*0.5) for t in arr_t]
106 #beta_a = beta_t + [ exp(1j*rho*t/2.65)*exp(-t*0.5)*(1-1/(t*1.5+1)) for t in arr_t]
107 F = [ 1-1/(t*1.5+1) for t in arr_t] #exp(-t*0.5)
108 #print(beta_t)
109 plt.plot(beta, label="beta_ij")
110 plt.plot(F, label="f(t)")
111 plt.yscale('log')
112 plt.legend(title=text, loc='upper right')
113 plt.xlabel(r"$t$", fontsize=18)
114 plt.ylabel(r"$n_{\{a\}}$", fontsize=18, rotation='horizontal', horizontalalignment='right')
115 plt.show()

```

Листинг Б.7 – n0.py

```

1 from scipy import integrate
2 from cmath import exp, sqrt, pi
3 import math
4 import R
5
6 def func(alpha, omega, omega_a, n_a0):
7     return lambda t: n_a0*math.exp(-2*omega**2*alpha*integrate.quad(lambda tau: 1/(1-math.exp(-2*alpha*omega**2*tau))*(-1+math.sqrt
    (1-2*omega_a**2*(1-math.exp(-2*alpha*omega**2*tau)))/(alpha*omega**2)), 0.01, t)[0])
8
9 def calc(d_t, alpha, omega, omega_a):
10     n_a0 = 1
11     n_a = func(alpha, omega, omega_a, n_a0)
12     n = list()
13     for t in ( x*d_t for x in range(int(1/d_t)) ):
14         n.append(n_a(t))
15     return n

```