

КВАНТОВЫЙ САМООРГАНИЗУЮЩИЙСЯ ИНТЕЛЛЕКТУАЛЬНЫЙ РЕГУЛЯТОР В РОБОТОТЕХНИКЕ: КВАНТОВЫЙ НЕЧЕТКИЙ ВЫВОД СО ВСТРОЕННЫМ КВАНТОВЫМ ГЕНЕТИЧЕСКИМ АЛГОРИТМОМ

Боровинский Виталий Викторович¹, Капков Роман Юрьевич²,
Николаева Алёна Валериановна³, Решетников Андрей Геннадьевич⁴,
Тятошкина Ольга Юрьевна⁵, Ульянов Сергей Викторович⁶

¹Директор;
ООО «ATOM»;
Россия, 127566, г. Москва, пр-д Высоковольтный, д. 21, стр. 5;
e-mail: vitali.borovinsky@yandex.ru..

²Предприниматель, преподаватель-исследователь;
Государственный университет «Дубна»;
Россия, 141980, Московская обл., г. Дубна, ул. Университетская, 19;
e-mail: info@kapkov.pro.

³Инженер математического моделирования;
ООО «ATOM»;
Россия, 127566, г. Москва, пр-д Высоковольтный, 21;
e-mail: NikolaevaAlena8989@gmail.com.

³Кандидат технических наук, доцент;
Государственный университет «Дубна»;
Россия, 141980, Московская обл., г. Дубна, ул. Университетская, 19;
Старший научный сотрудник;
Объединенный институт ядерных исследований;
Россия, 141980, Московская обл., г. Дубна, ул. Жолио-Кюри, 6;
e-mail: agreshetnikov@gmail.com.

⁴Кандидат технических наук, доцент;
Государственный университет «Дубна»;
Россия, 141980, Московская обл., г. Дубна, ул. Университетская, 19;
e-mail: tyatyushkina@mail.ru.

⁵Доктор физико-математических наук, профессор;
Государственный университет «Дубна»;
Россия, 141980, Московская обл., г. Дубна, ул. Университетская, 19;
Главный научный сотрудник;
Объединенный институт ядерных исследований;
Россия, 141980, Московская обл., г. Дубна, ул. Жолио-Кюри, 6;
e-mail: ulyanovsky46@mail.ru.

В задачах управления, поддержка свойства робастности функционирования сложного, слабо структурированного объекта управления (ОУ) за счет применения интеллектуальной системы управления (ИСУ) и технологий интеллектуальных вычислений, необходима для достижения цели управления в условиях риска и непредвиденных (или нештатных) ситуаций. С алгоритмической точки зрения эффективное решение актуальной проблемы обеспечения устойчивого функционирования ОУ в условиях неопределенности и сохранения робастности ИСУ означает, что в используемом алгоритме достижения цели управления выполняются следующие необходимые и достаточные (в общем случае антагонистические) условия: 1) минимум исходной информации о внешней среде (или возмущение, действующее на ОУ); 2) минимальный расход обобщенного полезного ресурса в ОУ и ИСУ. Рассматривается применение квантовых поисковых алгоритмов (КПА) в интеллектуальных системах управления (ИСУ), т.е. в системах, которые способны функционировать в непредвиденных ситуациях с гарантированным достижением цели управления



Статья находится в открытом доступе и распространяется в соответствии с лицензией Creative Commons «Attribution» («Атрибуция») 4.0 Всемирная (CC BY 4.0) <https://creativecommons.org/licenses/by/4.0/deed.ru>

(свойство робастности). Разработана интеллектуальная технология (ИТ) проектирования робастных баз знаний (БЗ) с применением квантовых эффектов самоорганизации в условиях непредвиденных ситуаций управления и риска. Описываются принципы построения, структура и практическое применение разработанной ИТ проектирования робастных БЗ в ИСУ, эффективно и надежно функционирующих в условиях риска и непредвиденных ситуаций управления на основе модели самоорганизации БЗ. Результат разработки заключается в обеспечении гарантированного достижения цели управления в непредвиденных (нештатных) ситуациях управления в реальном времени, как следствие применения квантового генетического алгоритма (КГА) управления в структуре самоорганизующейся ИСУ. Приведены примеры эффективного моделирования самоорганизации робастных БЗ в ИСУ динамически (глобально или локально) неустойчивыми существенно-нелинейными ОУ.

Ключевые слова: квантовый нечеткий вывод, квантовый поисковый алгоритм Гровера, квантовый генетический алгоритм, симулятор на классическом компьютере, квантовый интеллектуальный регулятор.

Для цитирования:

Квантовый самоорганизующийся интеллектуальный регулятор в робототехнике: квантовый нечеткий вывод со встроенным квантовым генетическим алгоритмом / В. В. Боровинский, Ю. Р. Капков, А. В. Николаева [и др.] // Системный анализ в науке и образовании: сетевое научное издание. 2025. № 4. С. 44-92. EDN: EEXSOF. URL: <https://sanse.ru/index.php/sanse/article/view/691>.

QUANTUM SELF-ORGANIZED INTELLIGENT CONTROLLER IN ROBOTICS: QUANTUM FUZZY INFERENCE WITH EMBEDDED QUANTUM GENETIC ALGORITHM

**Borovinsky Vitalii V.¹, Kapkov Roman Yu.²,
Nikolaeva Alyona V.², Reshetnikov Andrey G.³,
Tyatyushkina Olga Yu.⁴, Ulyanov Sergey V.⁵**

¹*General manager;*

LLC "ATOM";

21 Vysokovoltny Lane, Moscow, 127566, Russia;

e-mail: vitali.borovinsky@yandex.ru.

²*Businessman, teacher-researcher;*

Dubna State University;

19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;

e-mail: info@kapkov.pro.

³*Mathematical Modeling Engineer;*

LLC "ATOM";

21 Vysokovoltny Lane, Moscow, 127566, Russia;

e-mail: NikolaevaAlena8989@gmail.com.

³*PhD in Engineering sciences, associate professor;*

Dubna State University;

19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;

Senior Researcher;

Joint Institute for Nuclear Research;

6 Joliot-Curie Str., Dubna, Moscow region, 141980, Russia;

e-mail: agreshetnikov@gmail.com.

⁴*PhD in Engineering sciences, associate professor;*

Dubna State University;

19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia.

e-mail: tyatyushkina@mail.ru.

⁵*Grand PhD in Physical and Mathematical Sciences, professor;*

Dubna State University;

19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;
Chief Researcher;
Joint Institute for Nuclear Research;
 6 Joliot-Curie Str., Dubna, Moscow region, 141980, Russia;
e-mail: ulyanovsv46_46@mail.ru.

In control tasks, maintaining the robustness property of a complex weakly structured control object (CO) through the intelligent control systems (ICS) and intelligent computing technologies using is necessary to achieve the goals in conditions of risk and unpredictable (or abnormal) control situations. From an algorithmic view point, an effective solution for the actual issue of ensuring the stable OC functioning under conditions of uncertainty and also ICS robustness maintaining, means that the following necessary and sufficient (antagonistic in general case) conditions to the control goals achieving in the used control algorithm's, should be satisfied: 1) a minimum of external environment initial information (or disturbance on CO acting); 2) the minimum consumption of the generalized useful resources in the CO and ICS. The application of quantum search algorithms in ICS, i.e. in systems that are able to function in unpredictable situations with guaranteed achievement of the control goal (robustness property) is considered. The intelligent technology (IT) design of robust knowledge bases (KB) with using quantum effects of self-organization under contingencies of unpredictable control situations and risk is developed. The principles of construction, the structuration and practical application of the developed IT for the robust KB design in ICS which is effectively and reliably functioning in the conditions of risk and unpredictable situations based on the self-organization KB model are described. The result of this development is consisting in guaranteed ensuring of the control goal achievement for unpredictable (abnormal) real-time control situations, as a consequence of a quantum genetic algorithm (QGA) control application in the self-organizing ICS structure. An example of the robust self-organization KB effective modeling in an ICS for a dynamically unstable and essentially nonlinear CO is presented.

Keywords: quantum fuzzy inference, Grover's quantum search algorithm, quantum genetic algorithm, classical computer's simulation, quantum intelligent controller.

For citation:

Borovinsky V. V., et al. Quantum self-organized intelligent controller in robotics: quantum fuzzy inference with embedded quantum genetic algorithm. *System analysis in science and education*, 2025;(4):44-92 (in Russ). EDN: EEXSOF. Available from: <https://sanse.ru/index.php/sanse/article/view/491>.

Введение

В [1-7] разработана информационная технология (ИТ) проектирования робастных баз знаний (БЗ) на основе открытого и подробно описанного в [8] синергетического эффекта квантовой самоорганизации неточных БЗ в непредвиденных ситуациях. Рассматривается применение квантовых поисковых алгоритмов (КПА) в интеллектуальных системах управления (ИСУ), т.е. в системах, которые способны функционировать в непредвиденных ситуациях с гарантированным достижением цели управления (свойство робастности). Развитие ИСУ привело от зарождения самой идеи, до создания коллективов роботов, автомобилей с автопилотом и множества других систем промышленного применения, которые способны работать в нештатных и непредвиденных ситуациях в проектах «Индустрія 4.0 / 5.0». Следовательно, разработка корректного алгоритма проектирования робастности ИСУ является одной из актуальных задач современной теории и систем управления; одновременно данная проблема относится к сложной и слабо исследованной области разработки ИСУ, способных эффективно и надежно функционировать в условиях риска и непредвиденных ситуаций управления [4-7].

Используемый в квантовом алгоритме принцип минимума информационной энтропии гарантирует необходимое условие самоорганизации – минимум требуемой исходной информации в сигналах обучения; термодинамический критерий минимума новой меры обобщенного производства энтропии обеспечивает достаточное условие самоорганизации – робастность процессов управления. В условиях риска и непредвиденных ситуациях оптимизация БЗ по информационно-термодинамическим критериям с помощью квантового алгоритма самоорганизации обеспечивает инвариантное достижение цели управления в реальном времени с требуемым уровнем робастности

ИСУ. Примеры эффективного моделирования самоорганизации робастных БЗ в ИСУ динамически неустойчивыми существенно нелинейными ОУ в непредвиденных ситуациях показали эффективность разработанной квантовой сквозной ИТ.

Цель работы: Целью работы является разработка ИТ проектирования робастных БЗ с применением квантовых эффектов самоорганизации [4,7-9] в условиях непредвиденных ситуаций управления и риска.

Полученный результат: Результат разработки заключается в обеспечении гарантированного достижения цели управления в непредвиденных (нештатных) ситуациях управления в реальном времени, как следствие применения в структуре самоорганизующейся ИСУ квантового генетического алгоритма (КГА), встроенного в квантовый поисковый алгоритм КНВ самоорганизации неточных БЗ (см. ниже (1)).

Методологический базис: В настоящей работе (выступающей продолжением [4,8,9]) описываются принципы построения, структура и практическое применение разработанной ИТ проектирования робастных БЗ в ИСУ, эффективно и надежно функционирующих в условиях риска и непредвиденных ситуаций управления на основе модели самоорганизации БЗ, разработанной в [8].

1. Квантовый генетический алгоритм: выбор типа и вида корреляции в квантовом нечетком выводе

Основной задачей разработанного КГА является поиск оптимальных видов и типов корреляций между классическими состояниями коэффициентов усиления ПИД-регуляторов в соответствии с физической интерпретацией слабоструктурированной модели ОУ и извлечения скрытой в классических состояниях коэффициентов усиления квантовой информации, которая является дополнительным информационным ресурсом для формирования робастных БЗ с применением КНВ [10,11]. Приведены примеры эффективного моделирования самоорганизации робастных БЗ в ИСУ динамически неустойчивым существенно-нелинейным ОУ.

1.1. Интеллектуальные системы управления

Развитие ИТ проектирования ИСУ, основанное на применении мягких вычисления, обычно используют методологии нечеткой логики, эволюционных алгоритмов и нейронных сетей. Базисом развития систем управления послужили *PID*-регуляторы, которые зачастую, не справляются с задачей управления и плохо работают в непредвиденных ситуациях. Нечеткие регуляторы, позволяют частично расширить сферу применения ПИД регуляторов, за счет добавления правил функционирования и частично адаптировать систему. Использование генетических алгоритмов и нечеткой нейронной сети позволило полностью адаптировать систему, но для обучения такой системы требуются временные затраты, что в нештатных и непредвиденных ситуациях является критическим критерием. Моделирования оптимального сигнала позволяет создать частичную самоорганизацию в системе за счет обнаружения оптимальных траекторий коэффициентов усиления ПИД-регулятора.

На рис. 1 показана ИСУ с объединением нескольких нечетких регуляторов и КНВ, которая позволяет создать новое качество в управлении – самоорганизацию БЗ в режиме *online*.

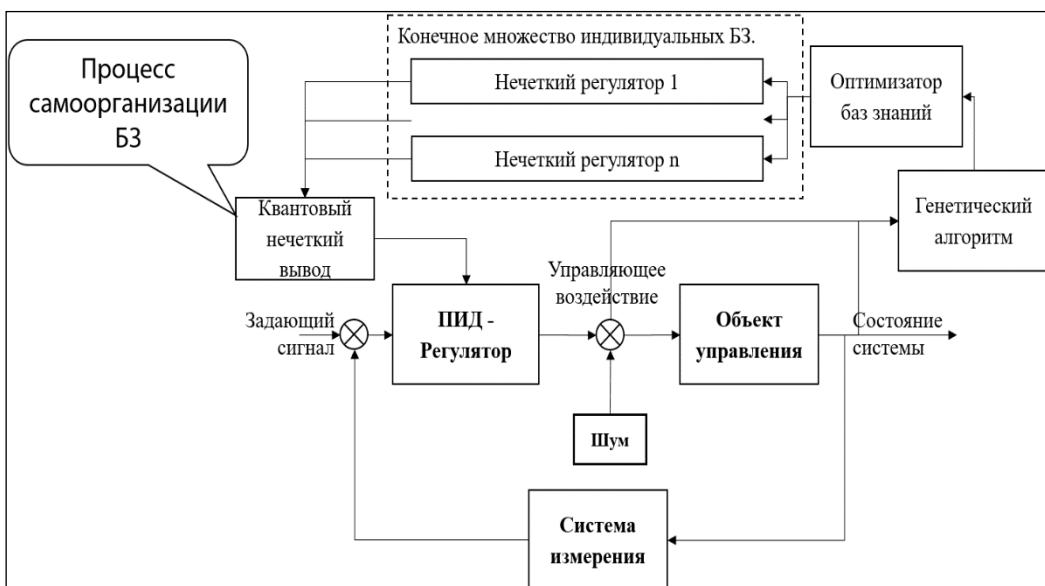


Рис. 1. Интеллектуальная система управления с квантовым нечетким выводом

Применение квантовых вычислений и, как частного примера КНВ, позволяет повысить робастность без затрат временного ресурса – в режиме реального времени.

1.2. Кvantовый нечёткий вывод

Главная проблема ИСУ заключается в том, что очень трудно спроектировать глобально хорошую и робастную структуру управления на все возможные случаи. Особенно, когда система работает в слабо предсказуемых ситуациях. Одно из лучших решений –формирование конечного числа БЗ нечеткого регулятора для множества фиксированных ситуаций управления. Цель квантового регулятора объединить БЗ, полученные при помощи оптимизатора баз знаний в самоорганизующиеся квантовые нечеткие регуляторы. Квантовый нечеткий вывод использует частные индивидуальные БЗ нечеткого регулятора, каждая из которых получена с помощью инструментария оптимизатора баз знаний.

Процесс проектирования квантовой алгоритмической ячейки включает матричную форму трех квантовых операторов: суперпозиции, запутанности и интерференции, которые являются частью структуры КПА. В общем виде структура квантовой алгоритмической ячейка (*QAG*) с применением квантового генетического алгоритма (*QGA*) описана в (1) в виде:

$$QAG = \left[(Int \otimes^n I) \cdot U_F \right]^{h+1} \cdot [QGA] \left[{}^n H \otimes {}^m S \right], \quad (1)$$

где I – тождественный матричный оператор; символ \otimes обозначает тензорное произведение; S равен I или H в зависимости от описания проблемы. Первая часть в проектировании уравнения (1) – это выбор типа оператора запутанного состояния U_F , который физически описывает качественные свойства функции.

Основным блоком такой ИСУ, является квантовый генетический поисковый алгоритм (КГПА, *QGSA – quantum genetic search algorithm*) (см. рис. 2).

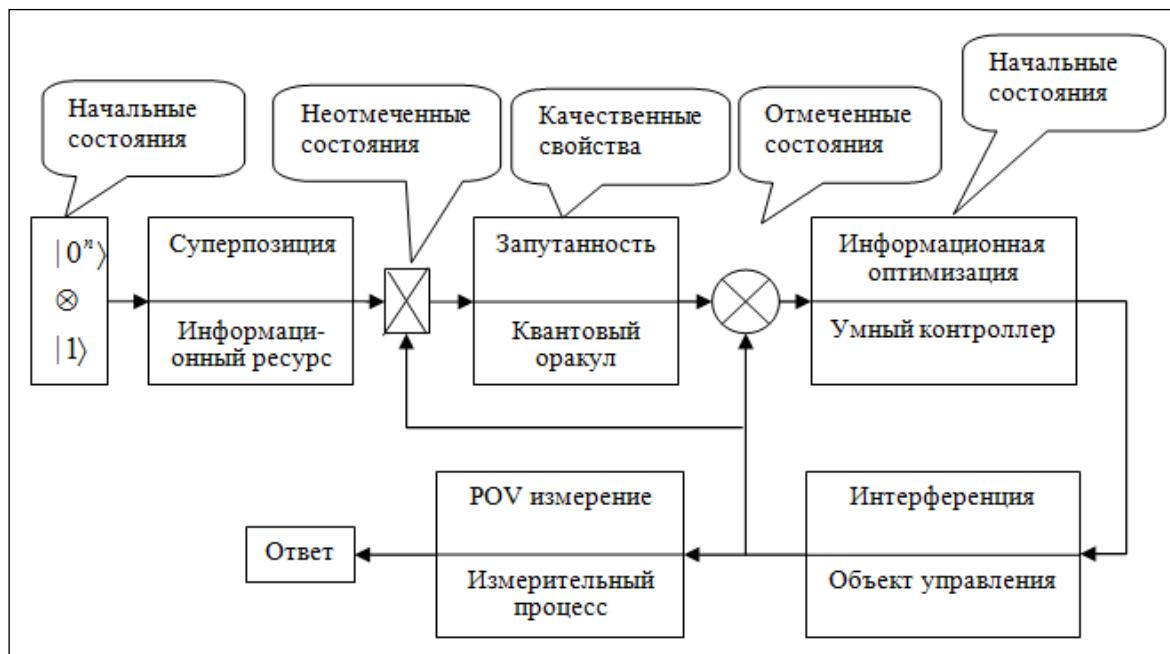


Рис. 2. Самоорганизующийся интеллектуальный квантовый поисковый алгоритм для интеллектуальной системы управления

Основные элементы структуры квантового поискового алгоритма (КПА, QSA – *quantum search algorithm*), как главного блока глобальной оптимизации, основанной на квантовых мягких вычислениях, представлены на рис. 3.

Формально структуру КГПА описывается следующим множеством логических операторов:

$$QGSA = \{ C, Ev, P^0, L, [\Omega, \chi, \mu]_{GA\text{-операторы}}, [Sup, Ent, Int]_{Q4\text{-операторы}}, \Delta \},$$

где C – алфавит для генетического кодирования индивидуума для конкретной задачи; Ev – функция пригодности; P^0 – начальная популяция; L – размер популяции; Ω – оператор отбора (селекции); χ – оператор скрещивания; μ – оператор мутации; Sup – квантовый оператор линейной суперпозиции; Ent – квантовый оператор запутывания (смешанное состояние); Int – оператор вывода; Δ – условие остановки, включающие такие критерии остановки, как оптимум заданной функции пригодности и минимум энтропии Шеннона/фон Неймана. Структура на рис. 2 – базовая модель интеллектуальной системы управления, описывает набор логических операторов КГПА.

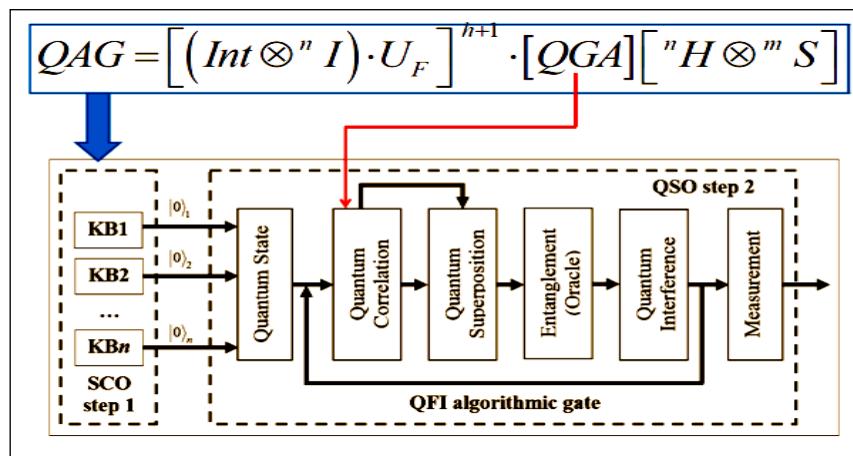


Рис. 3. Квантовая алгоритмическая ячейка (QAG) квантового нечеткого вывода

Комбинации логических операторов КГПА могут быть другими и отличаются от представленной формы, представляя различные варианты КПА. На рис. 3 представлена квантовая алгоритмическая

ячейка (*QAG*) квантового нечеткого вывода (КНВ). Такая ячейка может быть реализована как на классическом, так и на квантовом процессоре, а также может быть интегрирована в различные системы управления и встроенные интеллектуальные контроллеры.

Алгоритм КНВ для определения новых коэффициентов усиления ПИД-регулятора K (см. рис. 4), состоит из таких этапов, как нормализация, формирование квантового бита, после которого осуществляется подбор оптимальной структуры квантовой алгоритмической ячейки, выбирается состояние с максимальной амплитудой, осуществляется декодирование и на выходе получаем новый параметр K [7,8,11].



Рис. 4. Алгоритм квантового нечеткого вывода

На входе КНВ получает коэффициенты от сформированных заранее БЗ нечеткого регулятора [7] на основе оптимизатора БЗ на мягких вычислениях.

Следующим шагом осуществляется нормализация полученных сигналов [0,1] путем деления текущих значений сигналов управления на их максимальные значения (max K), которые заранее известны.

Пример: Формирование квантовых битов. Определяются функции плотности распределения вероятностей. Они интегрируются и из них получаются функции распределения вероятностей. Они позволяют определить «виртуальные» состояния сигналов управления для формирования суперпозиции с помощью преобразования Адамара из текущего состояния введённых сигналов управления. Используется закон вероятности: $p(|0\rangle) + (|1\rangle) = 1$, где $p(|0\rangle)$ – вероятность текущего реального состояния, а $p(|1\rangle)$ – вероятность текущего виртуального состояния. По текущему реальному состоянию из закона сохранения вероятностей определяется виртуальное состояние. Суперпозиция квантовой системы «реальное состояние – виртуальное состояние» имеет вид:

$$|\psi\rangle = \frac{1}{\sqrt{2}} (\sqrt{p(|0\rangle)}^* |0\rangle + \sqrt{(1-p(|0\rangle)}^* |1\rangle)$$

На рис. 5 показан процесс формирования квантовых бит для текущего реального состояния нормированного сигнала управления, описывающего коэффициенты усиления нечеткого ПИД-регулятора.

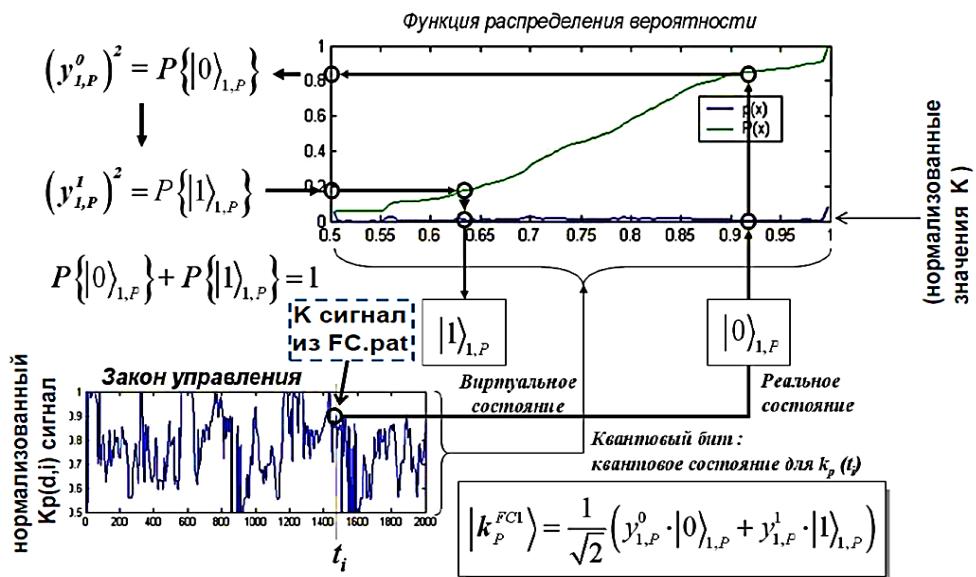


Рис. 5. Формирование суперпозиций состояний

На следующем этапе происходит выбор типа квантовой корреляции – операция построения запутанных состояний. Рассматривается три типа квантовой корреляции: пространственная, временная и пространственно-временная. Каждая из них содержит скрытую в спроектированных БЗ ценную квантовую информацию [11].

1.3. Квантовая корреляция и роль квантовой информации скрытой в классических состояниях

Квантовая корреляция рассматривается как физический вычислительный ресурс, позволяющий увеличить успешный поиск решений алгоритмически неразрешимых проблем. В нашем случае решение задачи обеспечения глобальной робастности функционирования ОУ в условиях непредвиденных ситуаций управления за счет проектирования оптимальной структуры и законов изменения коэффициентов усиления ПИД-регулятора классическими методами управления является алгоритмически неразрешимой проблемой [1 – 11]. Решение данной проблемы возможно на основе технологий квантовых мягких вычислений [11 – 15]. Выходные параметры ПИД-регуляторов рассматриваются как активные информационно взаимодействующие агенты, из которых формируется результирующая управляющая сила ОУ (см. ниже рис. 7 и 8). В многоагентной системе существует новый синергетический эффект, возникающий при обмене информацией и знаниями между активными агентами (*swarm synergistic information effect*) [7-9,13-16]. За счет квантового синергетического эффекта создается дополнительный информационный ресурс и многоагентная система (с учетом конечного множества БЗ), способна решать сложные динамические задачи по выполнению совместной работы.

Поставленная задача может не выполняться каждым элементом (агентом или базой знаний) системы в отдельности в разнообразных средах без внешнего управления, контроля или координации, но обмен знаниями и информацией позволяет совершать дополнительную совместную полезную работу коэффициентам усиления ПИД-регуляторов для достижения поставленной цели управления в условиях неопределенности исходной информации и ограничений на расход полезного ресурса [8,10,11].

Примечание. В частности, известно, что для систем управления с обратной связью, количество извлекаемой полезной работы W удовлетворяет неравенству $W_{\max}(t) = k \int_0^t T_{\min} I_c dt' \leq kT I_c$, где k – постоянная Больцмана на, $T_{\min}(t)$ интерпретируется как наименьшая достижимая температура системой во времени t при управлении с обратной связью, предполагая $T_{\min}(0) = T$, и I_c определяет количество информации Шеннона (перенос энтропии), извлекаемое системой из процесса измерения [11,17-19]. Физически синергетический эффект означает самоорганизацию знаний и создание дополнительного количества информации, которая позволяет совершить многоагентной системе полезную работу с минимумом потери полезного ресурса и при минимуме требуемой исходной

информации, без разрушения нижнего исполнительного уровня системы управления [8,13]. Совместно с информационно-термодинамическим законом интеллектуального управления (оптимальное распределение качеств управления «устойчивость – управляемость – робастность») проектируется ИСУ многоагентными системами, гарантирующая достижение цели управления в условиях неопределенности исходной информации и ограниченного полезного ресурса [8, 11, 12].

2. Разработанная модель самоорганизации и используемые результаты

В [1,12,15] задача проектирования свойства робастности (как самостоятельного свойства самоорганизации) ИСУ подробно изучена на основе новых видов интеллектуальных вычислений, таких, как мягкие и квантовые вычисления. Предложена модель КА самоорганизации ИСУ, базирующаяся на принципах минимума информационной энтропии (в “интеллектуальном” состоянии сигналов управления) и обобщенной термодинамической мере производства энтропии (в системе “объект управления + регулятор”). Основным результатом применения процесса самоорганизации является приобретение необходимого уровня робастности и свойства гибкости (адаптивности) воспроизводимой структуры. Отмечено, что свойство робастности (по своей физической природе) выступает в качестве составной части самоорганизации, а требуемый уровень робастности ИСУ достигается за счет выполнения принципа минимума производства обобщенной энтропии.

Принцип минимума производства энтропии в ОУ и системе управления [11] является физическим принципом оптимального функционирования с минимальным расходом полезной работы и лежит в основе разработки робастной ИСУ. Данное утверждение базируется на том, что для общего случая управления динамическими объектами оптимальное решение конечной вариационной проблемы определения максимума полезной работы W эквивалентно, согласно [11,20], решению конечной вариационной проблемы нахождения минимума производства энтропии S .

Таким образом, исследование условия максимума функционала $\max_{q_i,u}(W)$ (где q_i,u – обобщенные координаты ОУ и сигнал управления соответственно) эквивалентно, согласно [12,14], исследованию ассоциированной проблемы минимума производства энтропии, т. е. $\min_{q_i,u}(S)$.

Следовательно, в разработанной модели КА используемый принцип минимума информационной энтропии гарантирует необходимое условие самоорганизации – минимум требуемой исходной информации в сигналах обучения; термодинамический критерий минимума новой меры обобщенного производства энтропии обеспечивает достаточное условие самоорганизации – робастность процессов управления с минимальным расходом полезного ресурса.

2.1. Извлечение скрытой квантовой информации

Основная идея состоит в следующем: используя сигналы управления с первого и второго нечетких регуляторов, (см. рис. 1) строятся новые выходные значения коэффициентов усиления (вектора управления) в соответствии с выбранным типом корреляции.

Процесс оптимального извлечения ценной квантовой информации из классических состояний базируется на следующих четырёх фактах в квантовой теории информации:

- существует эффективный квантовый алгоритм сжатия данных;
- в квантовом состоянии присутствует «сцепленное» представление классической и квантовой информации;
- полная корреляция в квантовом состоянии представляет собой «смесь» классической и квантовой корреляций;
- присутствует скрытая (частично доступная извлечению) классическая корреляция в квантовом состоянии.

Каждая из корреляций характеризует связь между соответствующими сигналами управления обоих нечетких регуляторов или БЗ.

Для подсчёта новых k_p, k_l, k_D пространственной корреляции применяются следующие

формулы (1 и 2 индексы нечеткого регулятора):

$$\{k_P^1(t)k_D^2(t)k_D^1(t)k_P^2(t)\} \text{ – для } k_p;$$

$$\{k_D^1(t)k_D^2(t)k_I^1(t)k_I^2(t)\} \text{ – для } k_I;$$

$$\{k_I^1(t)k_I^2(t)k_P^1(t)k_P^2(t)\} \text{ – для } k_D.$$

Для подсчёта новых k_p, k_I, k_D временной корреляции применяются следующие формулы (1 и 2 индексы нечеткого регулятора):

$$\{k_P^1(t)k_P^1(t-\Delta t)k_P^2(t)k_P^2(t-\Delta t)\} \text{ – для } k_p;$$

$$\{k_D^1(t)k_D^1(t-\Delta t)k_D^2(t)k_D^2(t-\Delta t)\} \text{ – для } k_I;$$

$$\{k_I^1(t)k_I^1(t-\Delta t)k_I^2(t)k_I^2(t-\Delta t)\} \text{ – для } k_D.$$

Для подсчёта новых k_p, k_I, k_D пространственно-временной корреляции применяются следующие формулы (1 и 2 индекса нечеткого регулятора):

$$\{k_P^1(t)k_D^1(t-\Delta t)k_P^2(t-\Delta t)k_D^2(t)\} \text{ – для } k_p;$$

$$\{k_D^1(t)k_I^1(t-\Delta t)k_D^2(t-\Delta t)k_I^2(t)\} \text{ – для } k_I;$$

$$\{k_I^1(t)k_P^1(t-\Delta t)k_I^2(t-\Delta t)k_P^2(t)\} \text{ – для } k_D.$$

Для вычисления k_p пространственной корреляции используется алгоритм формирования суперпозиции запутанных состояний (см. рис. 6 и рис. 7).



Рис. 6. Выбор приоритетного квантового состояния

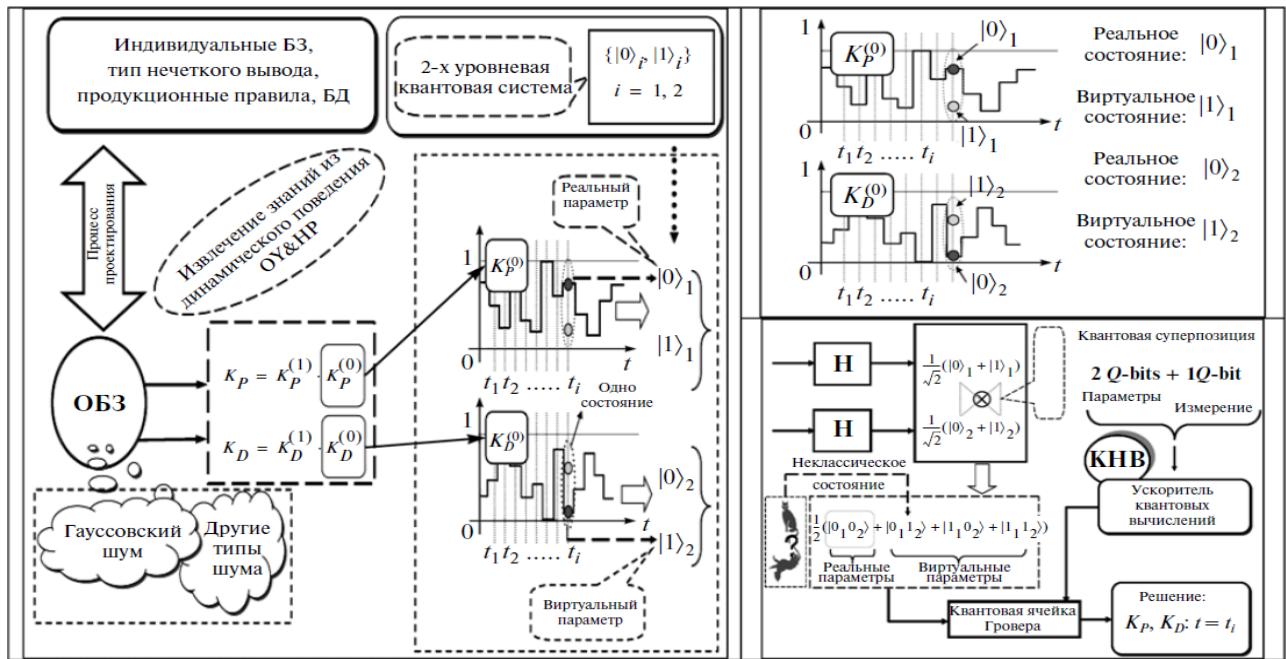


Рис. 7. Формирование и выбор компонент суперпозиции из различных сигналов управления и состояний (реальных и виртуальных)

Рассмотрим, например, согласно рис. 8, следующий тип квантовой корреляции: $\{k_p^1(t), k_d^1(t), k_p^2(t), k_d^2(t)\} \rightarrow k_p^{new}(t)$, где индексы 1 и 2 указывают принадлежность к соответствующим БЗ. Тогда квантовое состояние $|a_1 a_2 a_3 a_4\rangle = |k_p^1(t) k_d^1(t) k_p^2(t) k_d^2(t)\rangle$ рассматривается как коррелированное (entangled) состояние.

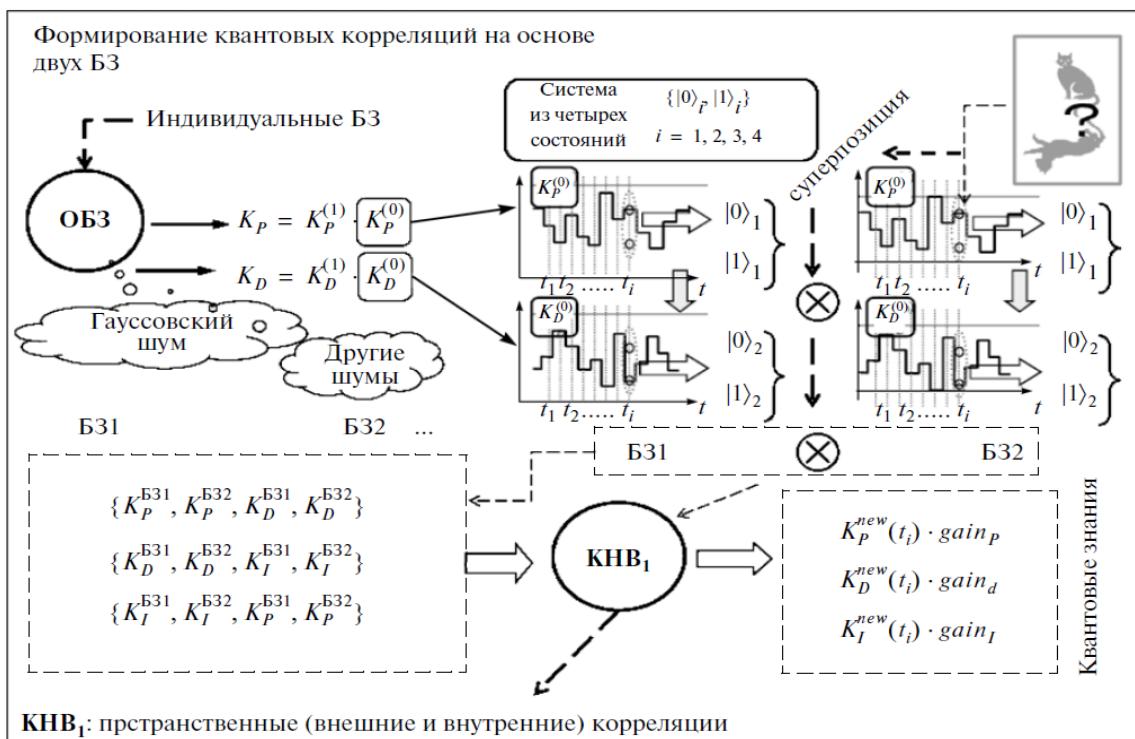


Рис. 8. Процесс формирования квантовой корреляции (пространственной, внешней и внутренней) между сигналами управления коэффициентами усиления K_p и K_d от двух БЗ

Графический интерфейс пользователя для формирования квантовых состояний в суперпозиции и их кодирование для выбранного типа квантовой корреляции показан на рис. 9.

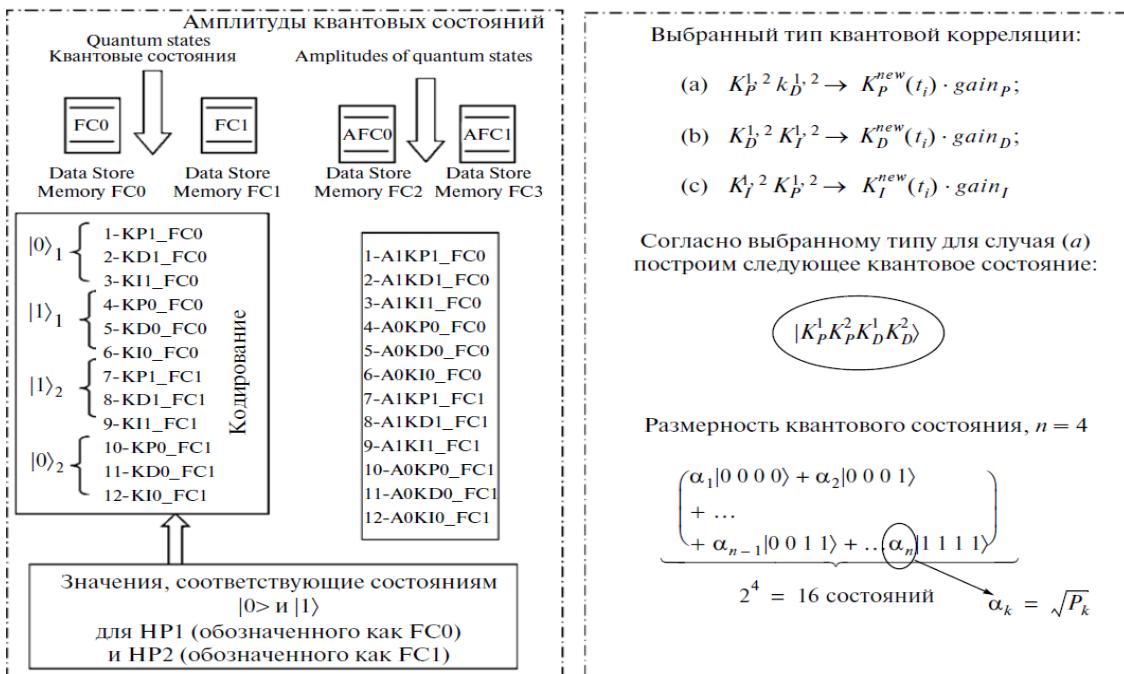


Рис. 9. Графический интерфейс пользователя формирования квантовых состояний в суперпозиции и их кодирование

Рисунок 10 содержит алгоритм вычисления пространственной корреляции (см. рис. 8) и процесс формирования «интеллектуального» состояния. На этом же рис. 10 показаны внутренняя и внешняя корреляции между сигналами управления коэффициентами усиления двух различных БЗ.

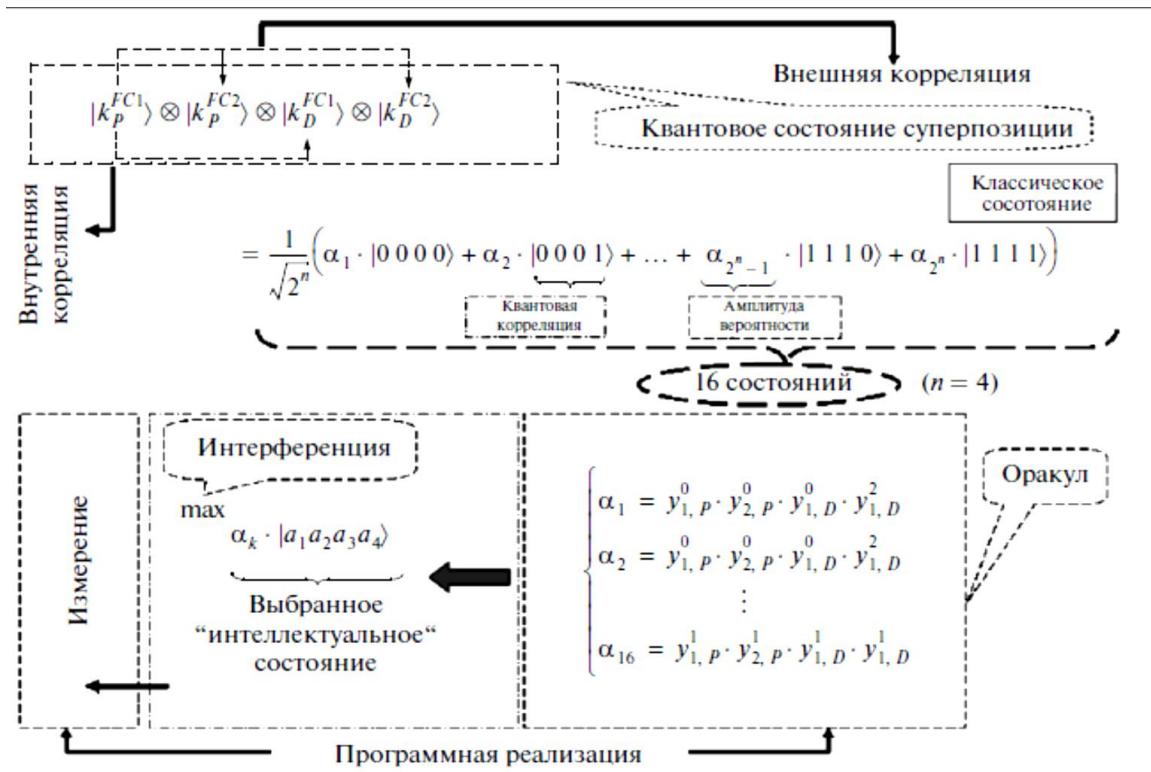


Рис. 10. Алгоритм вычисления пространственной корреляции и процесс формирования «интеллектуального» состояния

Графический интерфейс пользователя для формирования и вычисления «интеллектуального» квантового состояния по принципу максимума амплитуды вероятностей отражен на рис. 11.

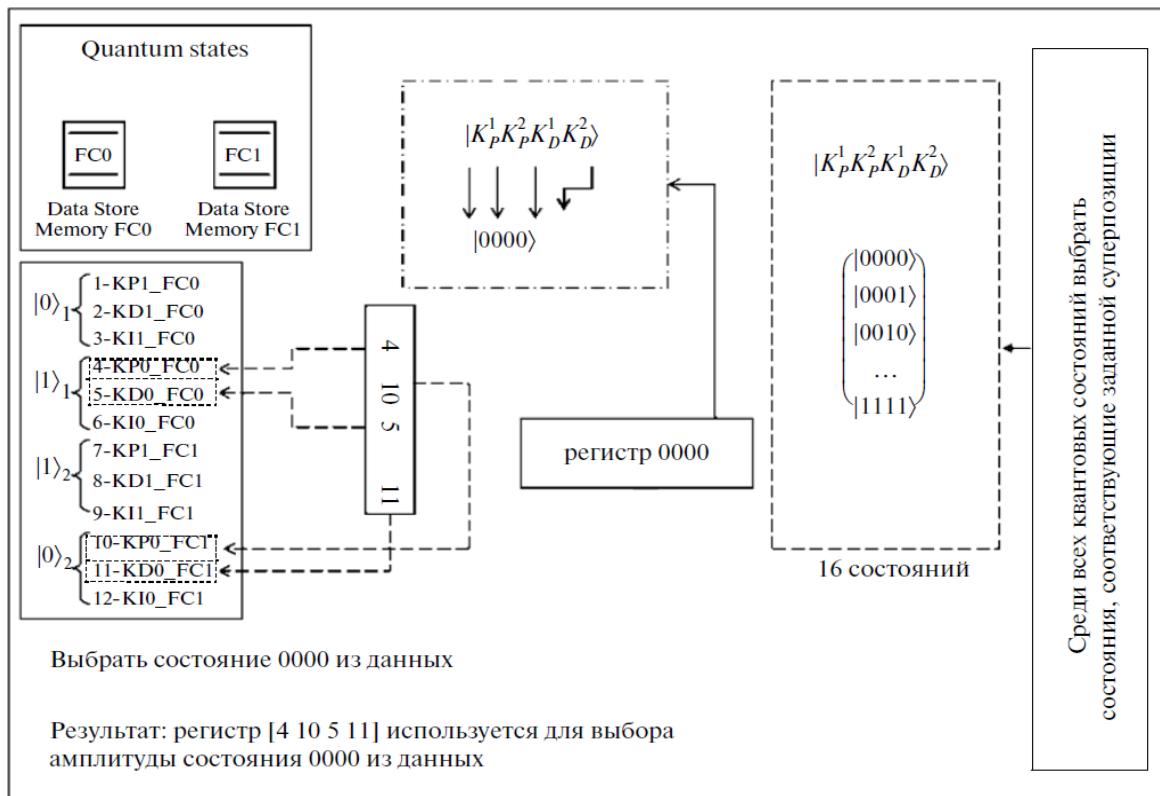


Рис. 11. Графический интерфейс пользователя для формирования и вычисления «интеллектуального» квантового состояния по принципу максимума амплитуды вероятностей

Алгоритм вычисления амплитуды вероятностей квантового состояния и его реализация в среде MatLab/Simulink представлены на рис. 12.

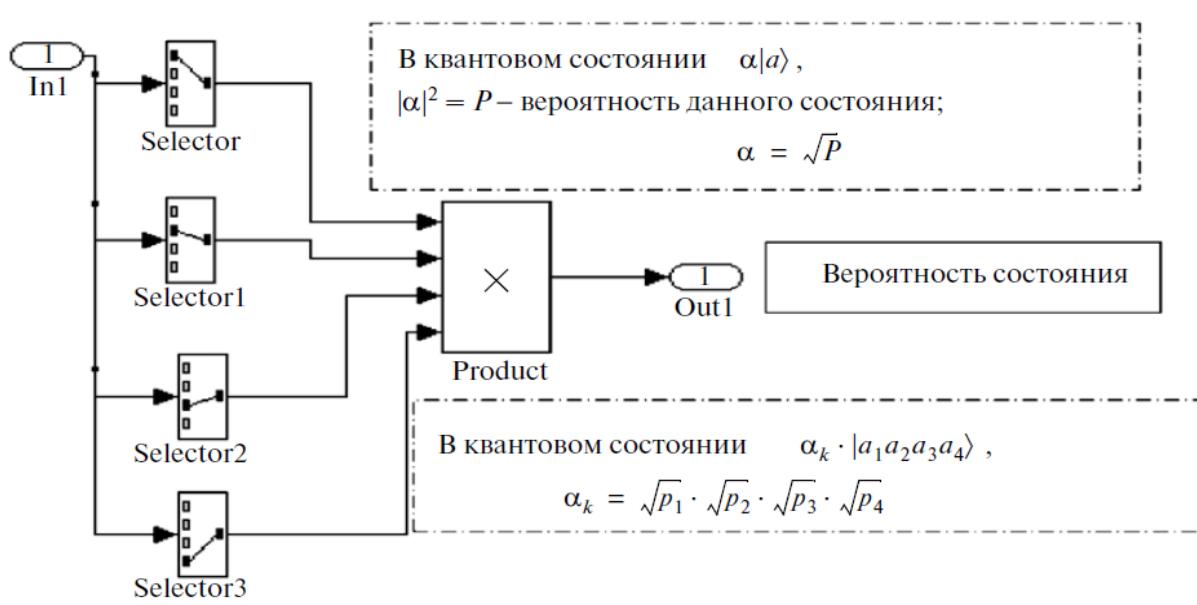


Рис. 12. Алгоритм вычисления амплитуды вероятностей квантового состояния и его реализация в среде MatLab/Simulink

Пример результатов работы блока измерений квантовых состояний с максимальной амплитудой вероятностей для трех коэффициентов усиления приведен на рис. 13.

Пример результатов блока измерений в непредвиденной ситуации управления:			
Состояния с максимальной амплитудой вероятностей	$K_P(t_i)$	$K_D(t_i)$	$K_f(t_i)$
1110	1101	1111	
1010	0101	1111	
1111	1111	1111	
1111	1111	1111	
1111	1111	1111	
1111	1111	1111	
1111	1111	1111	
1111	1111	1111	
1111	1111	1111	
1111	1111	1111	
1111	1111	1111	
1111	1111	1111	
1111	1111	1111	
1111	1111	1111	
1111	1111	1111	
1101	0111	1111	
0101	1111	1010	
0101	1111	1010	
0101	1111	1010	
1101	1111	1110	
1101	1111	1110	

| $K_P^1 K_P^2 K_D^1 K_D^2$ |

1 – виртуальное состояние,
 0 – реальное состояние
 $t_i, j = 0, 1, \dots, 21$
 – моменты времени
 управления

Рис. 13. Пример результатов блока измерений квантовых состояний с максимальной амплитудой вероятностей для трех коэффициентов усиления

В результате применения КНВ в структуре ИСУ осуществляется извлечение дополнительной (скрытой) квантовой информации, а ее использование дает возможность проектировать робастные сигналы управления в реальном времени из реакций НР на непредвиденные ситуации управления.

Отметим, что на рис. 13, показано, как новое состояние k_p строится из суперпозиции состояний: $|k_p^{FC1}\rangle \otimes |k_p^{FC2}\rangle \otimes |k_D^{FC1}\rangle \otimes |k_D^{FC2}\rangle$, которые присутствуют в выбранном запущенном состоянии. $|k_p^{FC1}\rangle$. Полученное состояние представляет из себя квантовое состояние из суперпозиции реального и виртуального состояний текущего k_p^{FC1} , и описывается следующим образом: $|k_p^{FC1}\rangle = \frac{1}{\sqrt{2}}(y_{1,P}^0|0\rangle_{1,P}y_{1,P}^1|1\rangle_{1,P})$. Применяя тензорное произведение между преобразованиями Адамара, в результате получается суперпозиция выбранных состояний (комбинации коэффициентов усиления). В предыдущем примере имеется 16 состояний.

Автоматизация выбора типа квантовой корреляции проводится при помощи квантового генетического алгоритма, выбор которого будет проиллюстрирован в следующем разделе.

3. Виды и операторы квантовых генетических алгоритмов (КГА)

Существует несколько различных видов КГА. Все они построены на комбинации квантовых и классических вычислений. Квантовые вычисления включают в себя квантово-генетические операторы, выполняющие генетические операции на квантовых хромосомах. Эти операторы называют *interference gates* (интерференционные гейты). Существует несколько операторов интерференции, но наиболее востребованным является *Q-gate* интерференции (вращения) [21-25].

3.1. Операторы КГА

Рассмотрим виды и особенности операторов КГА.

1. Обобщенный оператор *Q-gate* квантовой интерференции обозначается как gate $U(t)$:

$$U(t) = \begin{pmatrix} \cos(\delta\theta_j) & -\sin(\delta\theta_j) \\ \sin(\delta\theta_j) & \cos(\delta\theta_j) \end{pmatrix}$$

Применяя этот оператор, эволюция популяции есть результат процесса унитарных преобразований. В особенности, важны вращения (*rotations*), приближающие состояние хромосом к состоянию оптимальной хромосомы в популяции. Таким образом, гейт усиливает или уменьшает амплитуду кубитов или генов в соответствии с хромосомой с максимальной функцией пригодности (*fitness*): $f(x_1, x_2, x_3 \dots x_j)_i$ (*maximum*). Следовательно, эволюция квантового состояния определяется лучшими индивидуумами.

Примечание. В классическом ГА оператор выбора имитирует дарвиновский естественный отбор, улучшая популяции, продвигая особи с лучшей пригодностью и «наказывая» тех, кто имеет худшую производительность. В КГА выбор заменяется изменением всех особей на лучшую. Поэтому, когда популяция обновляется оператором вращения, популяция сходится, но обычно КГА попадает в локальные оптимумы, которые подвергаются преждевременной конвергенции (сходимости). Чтобы избежать этого КГА часто включают в себя либо рулетку, либо элитный выбор. Например, КГА с шагом выбора используется в улучшенном алгоритме кластеризации K - средних. Существуют ещё более экстремальные подходы, например, когда КГА включает в себя выбор и алгоритм имитации отжига, исключающий преждевременную конвергенцию. В других случаях шаг выбора включается, не прибегая к операторам, обычно используемым в ГА. Такой случай полуклассического ГА, где оператор селекции (выбора) стремится к максимальной пригодности посредством квантового подхода, например, используя алгоритм Гровера [10,25,26].

2. *Оператор квантовой мутации* (инверсия). В имитации ГА существует также квантовая версия классического оператора мутации. Гейт выполняет межкубитную мутацию j -го кубита, заменяя амплитуды квантовый гейтом Паули.

3. *Оператор квантовой мутации* (ввод). Этот гейт напоминает биологический механизм введения хромосом. Вставка хромосомы означает, что сегмент хромосомы был вставлен в необычную позицию на той же или другой хромосоме. Квантовая версия этого генетического механизма включает перестановку или обмен между двумя случайно выбранными кубитами (левый, правый). Например, предположим, что, учитывая следующую хромосому, выбирается случайным образом первый и третий кубиты.

4. *Оператор квантового перехода* (классический). Квантовый кроссовер моделируется на подобии классического алгоритма рекомбинации, используемого в ГА, но работает с амплитудами вероятности. И хотя квантовая версия мутации может быть реализована на квантовом компьютере, есть теоретические причины, препятствующие сделать кроссовер. Тем не менее, несмотря на теоретические ограничения, квантовая версия классического оператора кроссовера применяется во многих практических задачах оптимизации. В этих случаях поиск решения осуществляется с использованием квантового эволюционного «вдохновляющего» подхода. Вслед за этим оператор иллюстрируется для случая одноточечного кроссовера. В этом примере, если точка разреза произвольно выбирается, например, точка между первым и вторым положениями, тогда происходит обмен хромосомными сегментами.

5. *Оператор квантового перехода* (интерференция). Настоящий квантовый оператор выполняет кроссовер путем рекомбинации в соответствии с критерием, основанным на формировании диагоналей. В результате все особи смешиваются друг с другом, приводя к потомству.

3.2. Классификация квантовых эволюционных алгоритмов

Выделены два основных класса квантовых эволюционных алгоритмов: квантовый генетический алгоритм (КГА) и гибридный генетические алгоритмы (ГГА).

Схема КГА включает в себя следующие шаги:

- Инициализация квантовой популяции $Q(0)$
- Создаём $P(0)$, показатель каждого отдельного $Q(0) \rightarrow P(0)$

- Вычисляем $P(0)$ – классическое вычисление
- *While* (не состояние окончания) *do*
- Begin
- $t \leftarrow t + 1$
- *Q-gate* вращения
- *Q-gate* мутации
- Делаем оценку $Q(0) \rightarrow P(0)$
- Вычисляем $P(t)$ – классическое вычисление
- Завершение.

Схема ГГА включает в себя следующие шаги:

- Инициализация квантовой популяции $Q(0)$
- Создаём $P(0)$, показатель каждого отдельного $Q(0) \rightarrow P(0)$
- Вычисляем $P(0)$ – классическое вычисление
- *While* (не состояние окончания) *do*
- Begin
- $t \leftarrow t + 1$
- *Q-gate* вращения
- Оператор кроссовер
- *Q-gate* мутации
- Делаем оценку $Q(0) \rightarrow P(0)$
- Вычисляем $P(t)$ – классическое вычисление
- Завершение.

Оба квантовых алгоритма были протестированы [25] на примере применения их в задаче нахождения корней уравнения $f(x) = |x - \frac{\pi}{2} + \sin(x)|$.

4. Результаты моделирования

КГА в результате выполнения поставленной задачи показывает следующее (см. рис. 14).

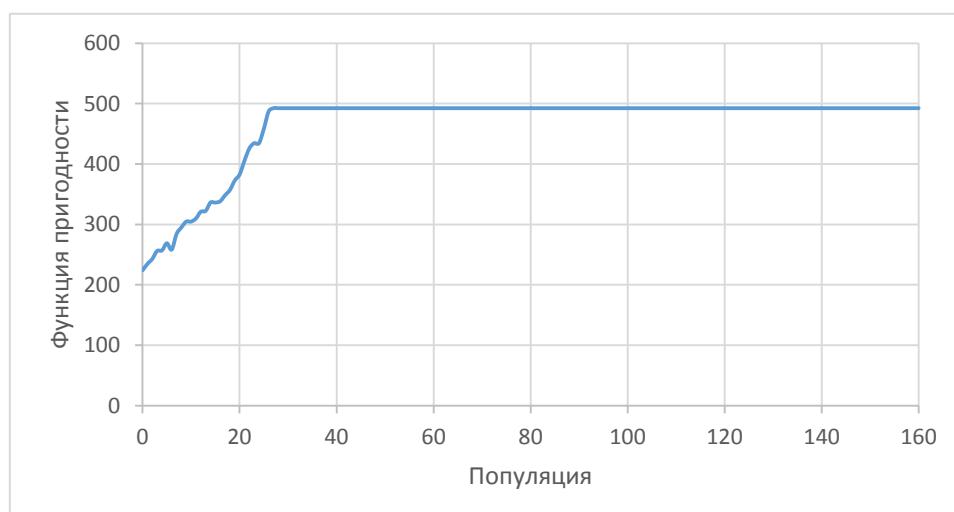


Рис. 14. Результат выполнения КГА

Видно, что после примерно 30 популяции значение функции пригодности перестаёт меняться. ГГА показывает следующие результаты (см. рис. 15).

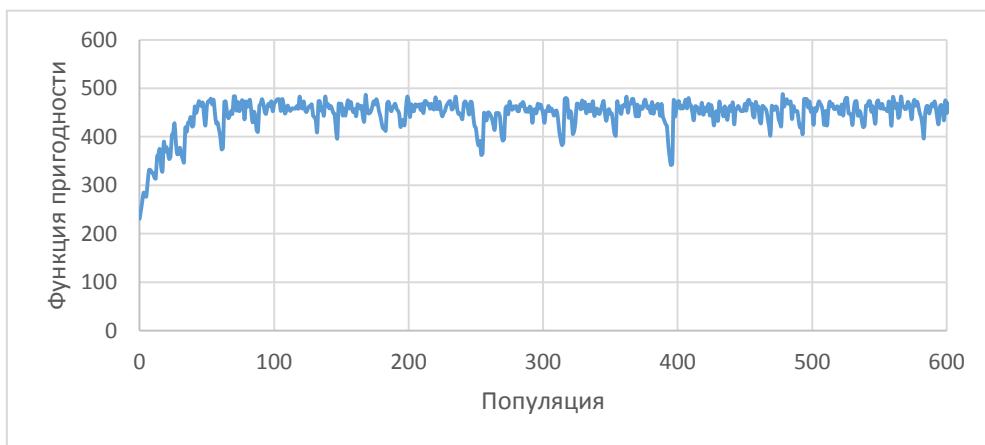


Рис. 15. Результат выполнения ГГА

КГА и ГГА алгоритмы можно рассматривать как классические методы оптимизации, основанные на принципах квантовых вычислений. Программы, реализующие такие методы, могут выполняться на цифровом компьютере, что подразумевает практические или теоретические трудности.

Примечание. В настоящее время одной из проблем в квантовом искусственном интеллекте является разработка истинных квантовых эволюционных алгоритмов и, следовательно, программ, которые в будущем могут работать на квантовом компьютере. Однако некоторые проблемы возникают, когда происходит перевод классического ГА в квантовую версию. Это даже можно назвать «парадоксом», т.к. ГА имеет сходство с квантовым алгоритмом Гровера: ГА – это параллельные методы поиска, хотя в обычных приложениях эта функция не реализована. Одной из основных проблем с алгоритмами КГА и ГГА является поиск метода для измерения популяции особей, но без разрушения состояния суперпозиции хромосом. Кроме того, на сегодняшний день ключевой вопрос заключается в том, как реализовать в квантовом компьютере оператор кроссовера. В то время, как мутация может быть легко проведена в квантовом компьютере с использованием гейта Паули, не ясно как выполнять кроссовер, используя для этого квантово-механические явления.

Одна из интересных идей была предложена в 2004 году [21], сделав первые шаги в реализации генетического алгоритма на квантовом компьютере [8,11]. Автором предложен настоящий квантовый эволюционный алгоритм, который можно назвать сокращенным квантовым генетическим алгоритмом (СКГА).

Алгоритм состоит из следующих этапов:

- Инициализация суперпозиции всех возможных хромосом;
- Оценка пригодности оператором F ;
- Применить алгоритм Гровера;
- Квантовый оракул;
- Применение диффузионного оператора Гровера G ;
- Сделать оценку.

Написав программу поиска решений уравнения для СКГА для той же задачи, что и КГА с ГГА результат выполнения стала матрица (см. рис. 16). Поиск решений выполняется за одну операцию. Результатом является матрица, где наибольшее значение имеет искомое решение (число 11).

```
[[0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]
 [0.078125]]
```

Рис. 16. Результат выполнения алгоритма СКГА

В КГА и ГГА эволюция является результатом унитарных преобразований, особенно вращений, приближающих состояние хромосом к состоянию оптимальной хромосомы с максимальной пригодностью. Поскольку эта процедура повторяется поколение за поколением, то результатом является быстрая сходимость к локальным оптимумам, имеющим место локальной ловушки.

Общая стратегия улучшения качества КГА заключается в использовании небольших усовершенствований алгоритма. Например, включая новые операторы, например, квантовое бедствие, возмущение или другие настроенные алгоритмы [22]. Но во многих случаях эти операторы полезны только в высокоспецифичных приложениях [23, 24].

Можно сделать вывод, что эволюция (оптимизация) квантовых эволюционных алгоритмов является результатом вращения квантового гейта, который вводит явление интерференции. Таким образом, особи корректируются или модифицируются, чтобы быть более похожими на лучшую особь в популяции. Получается, что популяция подвергается более низкому дарвиновскому давлению отбора. Однако в ГА после оценки особей алгоритм, имитирующий выбор (например, оператор выбора родителей) заменит старую популяцию $P(t)$ на новую популяцию $P(t+1)$ особей. Поскольку особи выбираются в соответствии с их значением пригодности, то популяция решений развивается через дарвиновскую эволюцию, но с большим давлением отбора, чем КГА и ГГА. Следовательно, ГА заменяет устаревшие стратегии инновационными стратегиями, представленными потомством. Аналогично, когда кроссовер включен как шаг в алгоритме ГГА его производительность обычно становится ближе к ГА.

Преимущество КГА в том, что они требуют меньше хромосом, чем ГА. В теории, в истинном квантовом генетическим алгоритмом (СКГА) можно считать, что популяция составлена из одной хромосомы в состоянии суперпозиции [10,21]. Этот факт может быть странным, но не с точки зрения квантовой механики. Более того, алгоритм СКГА демонстрирует, что с использованием квантовых вычислений стратегия генетического поиска становится ненужной. Эволюция происходит в одном поколении [25].

4.1. Анализ и сравнение результатов моделирования системы «движущаяся каретка – перевернутый маятник»

Проведя исследования с различными типами корреляции, было определено, что наилучший результат показывает пространственно-временная корреляция, обнаруженная ранее ручным методом в [7,8,16]. Запустив написанный КГА на несколько тысяч поколений (см. рис. 17), можно увидеть, что около 70% на лучшие показатели у пространственно-временной корреляции. $Q\text{-}S\text{-}T$ – пространственно-временная корреляция, $Q\text{-}T$ – временная, $Q\text{-}S$ – пространственная. Временная и пространственная корреляции имеют примерно одинаковый процент.

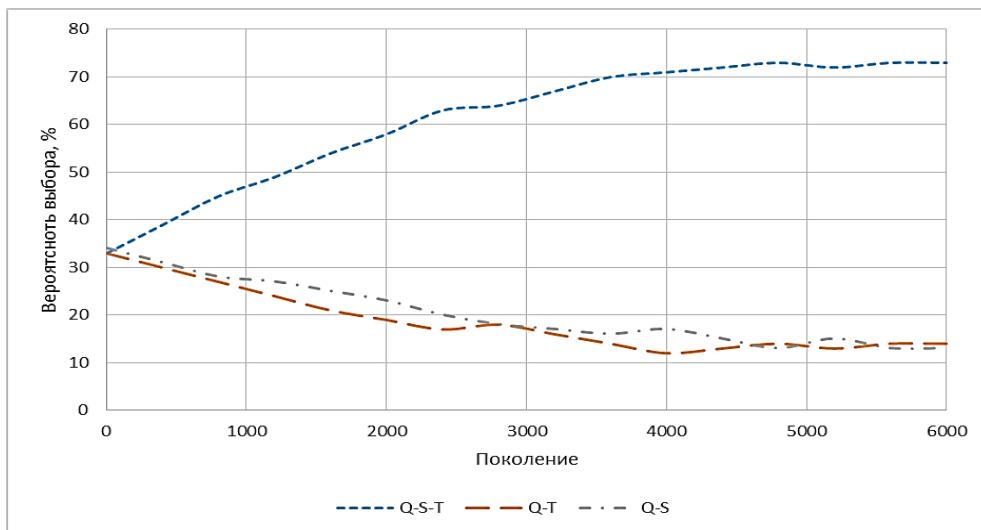


Рис. 17. Результат выполнения квантового генетического алгоритма

Стоит отметить, что после 5000 поколений в данном случае изменение вероятностей уже не происходит. Запустив КГА 200 раз, можно заметить, что вероятность выбора пространственно-временной корреляции опустилась до 60% (см. рис. 18).

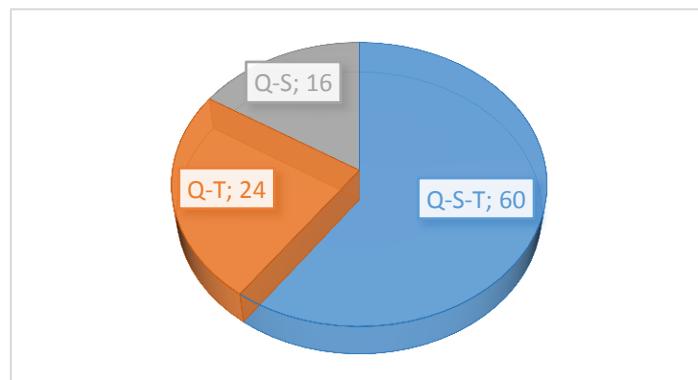


Рис. 18. Результат выполнения квантового генетического алгоритма 200 раз

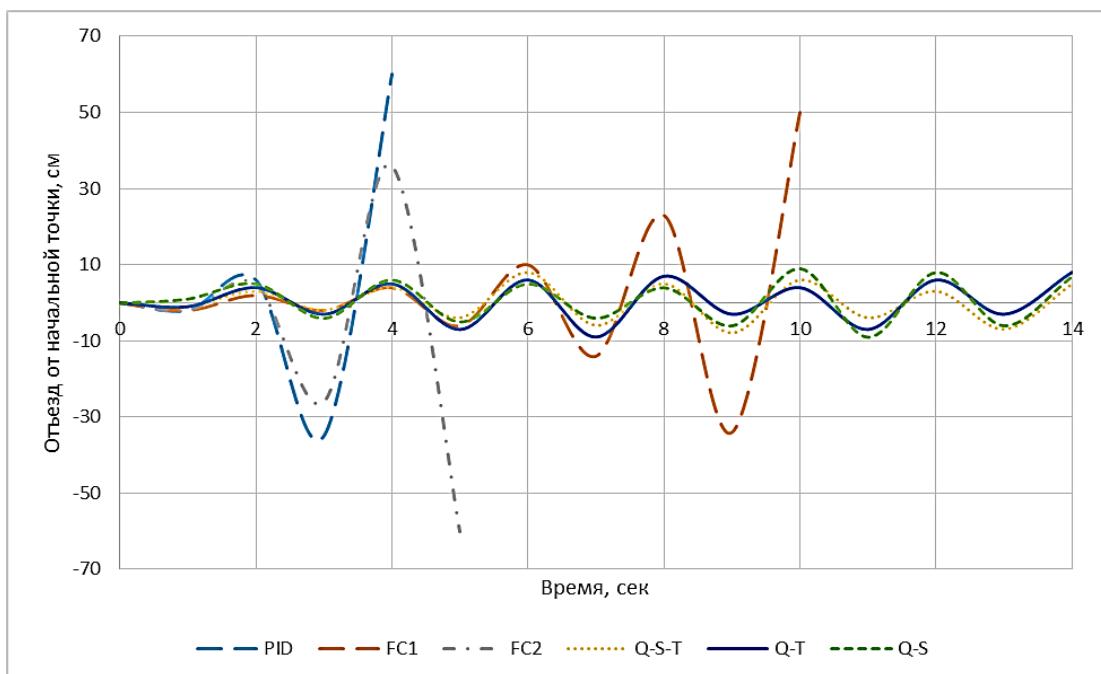


Рис. 19. Сравнение поведения каретки перемещения для различных регуляторов

После этого шага было смоделировано поведение различных алгоритмов на примере системы «движущаяся каретка-перевёрнутый маятник» (см. рис. 19).

Как видно на графике, *PID*-регулятор и нечёткий регулятор не справляются с заданной задачей из-за появления непредвиденной ситуации. Все три квантовые алгоритмы справились с ситуацией, а пространственно-временная корреляция показала наилучшие результаты.

5. Квантовый симулятор на основе квантового нечеткого вывода и квантового генетического алгоритма

Основная цель разработки симулятора – образовательные цели. Обучающиеся должны иметь возможность провести лабораторную работу, наблюдая поведение маятника при применении различных алгоритмов интеллектуального управления с разными параметрами: использование только *PID*-регулятора, добавление нечеткого регулятора в интеллектуальную систему управления (ИСУ), использование генетического алгоритма и нейронной сети, использование квантового генетического алгоритма. Особо заинтересованные читатели должны иметь возможность углубиться в техническую сторону реализации симулятора для реализации собственных алгоритмов, изучения уже реализованных или внесения различных улучшений в проект.

Симулятор интересен тем, что покрывает много областей, требуемых для его реализации. Также есть много различных путей развития: усовершенствование 2D модели или вообще реализация в 3D, управление маятником в режиме реального времени (изменение параметров маятника, добавление различных шумов), сделать симулятор более универсальным для простого создания симуляции других задач на основе подготовленного проекта. Симулятор охватывает разные направления и может применяться не только на курсе робототехники.

5.1. Выбор средств разработки

Доступ к симулятору должен быть максимально простой, поэтому было решено разработать веб-приложение – клиент-серверная программа, в которой пользователь взаимодействует с сервером при помощи браузера [10,25]. Оно состоит из нескольких частей: клиентская (или *frontend*) и серверная часть (*backend*).

Симулятор – это не типичное веб-приложение. Большая часть работы на серверной части – с математикой. Необходимо по множеству заданных и вычисляемых параметров рассчитывать положение каретки, угол наклона маятника в пространстве. По этому критерию выделяется один язык - *Python* – это лучший язык для выполнения каких-то научных работ. Содержит огромное количество библиотек, в т.ч. *NumPy* – фундаментальный пакет для научных исследований, который содержит в себе линейную алгебру, преобразование Фурье и многое другое [21], также понадобится функция линейного квадратичного регулятора, вычисляющего оптимальное управление, которое минимизирует квадратичную ошибку [1].

Все сгенерированные значения должны быть сохранены. Для этого понадобится СУБД. Поначалу приложение будет генерировать небольшое количество значений, но на парах, во время проведения лабораторных работ нагрузка может серьезно возрастать. СУБД для проекта выбрана *MySQL*, однако при реализации симулятора в режиме реального времени возможно скорости работы *MySQL* будут не хватать и для такой симуляции будет добавлена *Redis*.

Веб-фреймворк содержит набор библиотек, и обработчик, при помощи которого можно создавать пользовательский код для реализации веб-приложения. Существуют несколько распространённых фреймворков: *Django*, *Flask*, *Tornado*, *Pyramid*. *Django* самый крупный веб-фреймворк, предоставляет множество пакетов, шаблонов, утилит из коробки, стремится помочь быстро помочь строить сложные веб-приложения. Имеет большое активное сообщество. Большинство веб-приложений, написанных на *python* используют *Django*. *Django* – самый универсальный вариант, который и будет применяться в этом проекте. В официальной документации есть огромное количество примеров, найти ответ можно практически на любой вопрос. У веб-фреймворка большое сообщество, где можно с уверенностью получить ответ на любой вопрос.

Django реализует подход *model-view-controller (MVC)* (или больше *MVT – model-view-template*) [21]. *Model*: содержит объектно-ориентированное отображение (*object relational mapping – ORM*), которое взаимодействует между моделями, описанными как класс в *python* и реляционной базой данных. *View*: система обработки *HTTP*-запросов с системой веб-шаблонов. *Template*: отображает данные в *HTML*.

В последующих частях будет ещё уделено внимание развёртыванию приложения на веб-сервере.

5.2. Проектирование симулятора перевёрнутого маятника

Проектирование приложение очень важная часть разработки. Именно на этом этапе формируется полная картина возможностей системы.

Для пользователя должен иметься доступ к созданию маятника, а также просмотр истории создания маятников. Для создания маятника необходимо указать все параметры, используемые для работы алгоритма симулирования системы. Для просмотра истории маятников необходимо хранить сгенерированные координаты в базе данных.

Согласно методологии *MVT*, используемой в ранее выбранном веб-фреймворке *Django*, пользователь взаимодействует с шаблонами, которые связаны с представлениями, а те, в свою очередь, связаны с моделью. В симуляторе должно быть 5 страниц:

- главная страница с описанием проекта и наличием навигационного меню к остальным разделам;
- страница истории проведённых симуляций с поиском по различным критериями;
- страница создания новых симуляций с формой заполнения параметров;
- страница с описанием выполненных и будущих шагов развития проекта;
- страница просмотра самой симуляции.

Демонстрация пошаговой работы системы изображена на диаграмме последовательности (см. рис. 20).

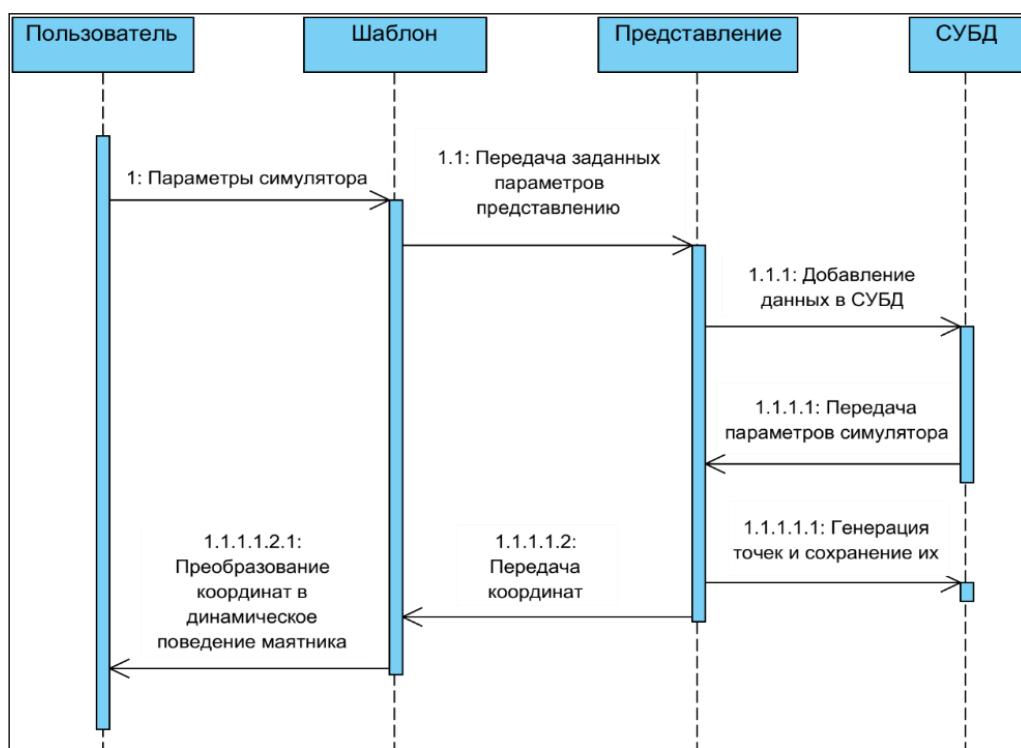


Рис. 20. Диаграмма последовательности работы системы

Пользователь передаёт через шаблон параметры симулятора, которые получает представление. Представление записывает заданные параметры в БД и начинает генерировать координаты по

полученным параметрами симулятора. Эти точки также сохраняются в БД, а затем передаются шаблону. Визуализация симуляция интеллектуальной системы управления перевёрнутым маятником осуществляется при помощи *JavaScript*. Веб-приложение, написанное на *Django*, состоит из нескольких частей:

Виртуальная окружающая среда. *Python* импортирует различные библиотеки, которые хранятся в папке, в которую он установлен. Но существует соглашение не устанавливать туда все библиотеки, т.к. могут возникать различные конфликты между разными приложениями, которые требуют, например, разные версии библиотеки. Чтобы решить эту проблему внутри каждого проекта необходимо создать *virtualenv*, куда и будут устанавливаться все библиотеки, использующиеся в текущем проекте.

Главный файл всего проекта – *manage.py*, который импортирует все настройки проекта, проверяет установлен ли *Django*.

Главная папка проекта – содержит файл с импортированием модуля *WSGI*. Файл с описание всех ссылок. В нём описываются именно ссылки на сами приложения. В нашем случае по умолчанию ссылка на *admin* и, добавленная разработчиком, *pendulums* – созданное приложение. Также содержит все настройки проекта:

- установленные приложение (туда входит уже готовые в веб-фреймворке приложение админ панели, аутентификации, сессии, сообщений, статический файлов), а также туда записываем, если разработчиком было создано своё приложение (в данном случае – *pendulums*);
- используемые шаблоны. По умолчанию используется *DjangoTemplates*, но в случае необходимости могут быть использовано, например, *Jinja2*;
- путь к файлу с *route* (ссылками приложения);
- путь к файлу с *WSGI*;
- подключаемый движок базы данных (в зависимости от того, какая СУБД используется в проекте), а также *host*, *port*, *user*, *database name* для подключения к СУБД;
- список различных проверок паролей;
- язык проекта;
- часовой пояс;
- путь к статическим файлам;
- различные пользовательские настройки по мере необходимости.

Созданное пользовательское приложение (*pendulums*) содержит следующее:

- файл с описанием тестов проекта;
- файл со списком ссылок данного приложения. В данном случае по ссылке */pendulums* будет располагаться весь список уже созданных маятников с указанием их параметров, на любой из которых можно нажать и посмотреть его работу. По ссылке *pendulums/create* можно будет создать новый маятник, после создания пользователя отправит на страницу с визуализацией работы. По ссылке *pendulums/<int:pk>*, где *<int:pk>* – уникальный идентификатор маятника, по которому будет отправлен запрос к БД, из которой необходимо получить координаты маятника с таким *ID*;
- файл с описание форм проекта;
- файл с конфигурацией приложения, по умолчанию туда входит только название приложения;
- файл, в котором необходимо указать, какие модели будут доступны для изменения в административной панели;
- папка, содержащая статические файлы проекта. Это могут быть как стили страниц, так и различные изображения, подключаемые библиотеки;
- папка миграций. Миграция – это изменение БД. Если разработчик вносит изменения в модели, то после этого необходимо провести миграцию. При помощи встроенных средств изменить структуру СУБД, чтобы она стала одинаковая со структурой модели. В папке содержатся все изменения БД, что очень удобно, похоже на систему контроля версий;
- файл с описанием моделей проекта;

- файл с представлениями;
- папка с шаблонами (о последних 3-х пунктах более подробно будет рассказано дальше, т.к. они содержат всю основную логику приложения).

Файл с описанием моделей *models.py* содержит объектно-реляционное отображение базы данных в виде классов. В приложении это класс *Pendulums*, который содержит все изменяемые параметры маятника и несколько параметров окружающей среды:

- шаг времени (*dt*) – при рисовании от этого показателя зависит количество точек в секунду. Отметим, что при очень большом количестве точек ожидание после создания маятника может несколько затянуться. Выставлять значения меньше 0.01 не имеет смысла, т.к. это уже будет 100 *fps*, увеличение его ещё больше незаметно на мониторе, а после 200 *fps* это влияет только на скорость вывода графического изображения (она будет гораздо ниже);
- время моделирования (*time*) – сколько в секундах будет длиться моделирование положения маятника. Без сильных шумов вероятность того, что маятник перестанет работать нулевая, поэтому нет необходимости моделировать более 30 сек. С добавлением симуляции в реальном времени моделирование может стать бесконечным по времени, однако в такой ситуации, при сохранении возможности просматривать историю БД может очень быстро увеличиваться в размерах;
- масса каретки и маятника (*M*);
- масса маятника (*m*);
- постоянная момента двигателя (*Km*);
- передаточное число (*Kg*);
- сопротивление (*R*);
- радиус движения (*r*);
- высота маятника (*l*);
- инерция (*I*);
- гравитация (*g*);
- напряжение (*Vsat*);
- алгоритм управления (*PID*, генетический алгоритм, квантовый генетический алгоритм);
- дата создания.

Класс *Coordinates*, который включает атрибут положения центра каретки, а также угла отклонения маятника, из которого при помощи *sin* и *cos* можно получить в градусах отклонение по оси *X* и *Y*. Этот класс связан с классом *Pendulums* при помощи внешнего ключа *Pendulums.id = Coordinates.pendulum_id*.

В файле с описание представлений *view.py* происходит вся основная логика программы. В нём происходит получение параметров от пользователя, вычисления поведения маятника по этим параметрам, запись их в БД и передача этих параметров шаблонам. Шаблон в свою очередь выводит переданные представлением параметры на экран в определённом формате. В файле представлений было описано несколько классов.

IndexView, который сработает при запросе от пользователя вернуть данные по ссылке */pendulums*. Класс возвращает 10 последних смоделированных маятников, передаёт эти данные шаблону с названием *index.html*. В шаблоне в виде табличке выводятся все данные, полученные от представления (см. рис. 21).

Create new pendulum														
#	date	dt	time	mass	mass of pendulum	torque constant	gear ratio	resistance	drive radius	length pendulum	inertia	gravity	voltage	
43	29/05/2018 23:52	0.050	10.00	0.600	0.300	2.00	0.010	6.00	0.010	0.300	0.0060	9.81	20.00	
42	20/05/2018 20:55	0.004	10.00	0.600	0.300	2.00	0.010	6.00	0.010	0.300	0.0060	9.81	20.00	
41	20/05/2018 20:55	0.003	10.00	0.600	0.300	2.00	0.010	6.00	0.010	0.300	0.0060	9.81	20.00	
40	20/05/2018 20:51	0.005	10.00	0.600	0.300	2.00	0.010	6.00	0.010	0.300	0.0060	9.81	20.00	
39	20/05/2018 20:50	0.001	10.00	0.600	0.300	2.00	0.010	6.00	0.010	0.300	0.0060	9.81	20.00	
38	19/05/2018 23:46	0.050	10.00	0.600	0.300	2.00	0.010	6.00	0.010	0.300	0.0060	9.81	20.00	
37	19/05/2018 22:43	0.010	10.00	0.600	0.300	2.00	0.010	6.00	0.010	0.300	0.0060	9.81	20.00	
36	19/05/2018 22:42	0.015	10.00	0.600	0.300	2.00	0.010	6.00	0.010	0.300	0.0060	9.81	20.00	

Рис. 21. История созданных симуляций

DetailView, который срабатывает при запросе от пользователя вернуть данные по ссылке */pendulums/id*, где *id* – уникальный идентификатор маятника. Представление делать запрос к БД, от которой получает список всей информации о маятнике с заданным *id*, в том числе и все данные из связанных таблиц, что очень удобно. Дальше эти данные получает шаблон *detail.html*, в котором создаётся *canvas*, в котором и осуществляется рисование. При помощи *JavaScript* происходит подготовка данных, полученных из БД, задаётся размер маятника и каретки, скорость рисования (*dt* * 1000). По данным из БД находится *sin* и *cos*, по которым определяются точки *x* и *y* в координатах *canvas*, которые и используются для рисования. Для рисования создана функция *draw*, которая выполняется столько раз, сколько точек имеет этот маятник и выполняется с заданной скоростью рисования. Алгоритм рисования, следующий: из центральной каретки точки влево на заданное количество – первая координата, вправо – вторая. У маятника одна координата находится тоже в центре каретки, а вторая находится по данным из БД.

В зависимости от заданных параметров веб-приложение генерирует от нескольких тысяч точек. Все они последовательно обрисовываются. Несколько положений маятника показано на рис. 22.

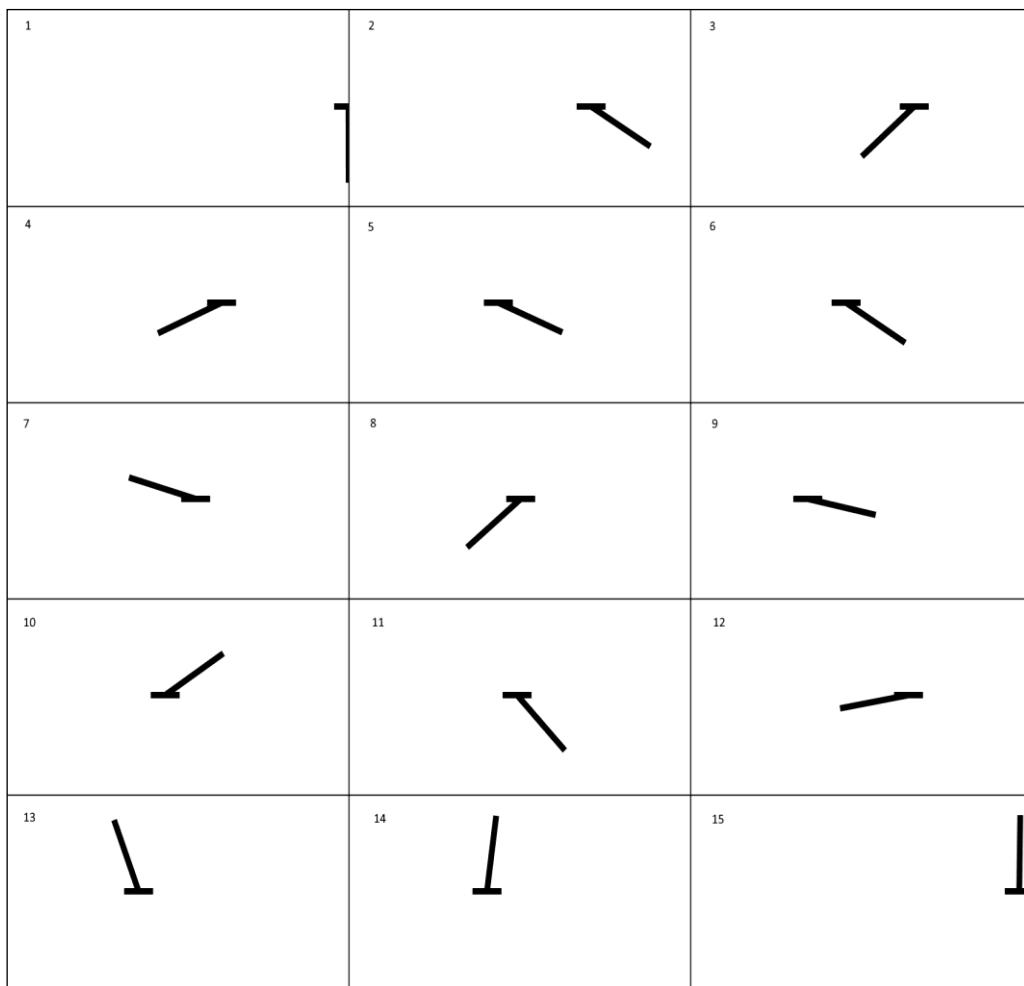


Рис. 22. Визуализация выполнения алгоритма

Каретка начинает раскачивать маятник и удерживает его в верхнем вертикальном положении.

PendulumView – класс, в котором находится вся основная логика программы. Шаблон создаёт форму, которую пользователь может изменить, чтобы смоделировать маятник с другими параметрами, после нажатия на кнопку сохранить данные получает этот класс. В нём создаётся объект класса *PendulumGen*, в котором и происходит генерирования точек расположения маятника в пространстве при помощи математических формул.

Пример: Применение ОВЗ на квантовых вычислениях с обучающим физическим сигналом с ОУ. На рис. 23 представлены результаты проведенного математического моделирования и физического эксперимента регистрации обучающего сигнала с реальных физических объектов (автономные роботы) в нештатной ситуации управления. В качестве нештатной ситуации выбрана ситуация с увеличением задержки сигнала обратной связи в три раза. Исследование подтвердило существование синергетического эффекта самоорганизации в процессе формирования робастной БЗ из спроектированных, не робастных БЗ. При этом дополнительный информационный ресурс для управления основан на извлечении скрытой в классических состояниях квантовой информации. Метод проектирования ИСУ позволил достигать глобальной робастности в режиме реального времени за счет использования новых типов вычислений, таких как квантовые и мягкие вычисления, при этом использовались вычислительные ресурсы классического процессора.

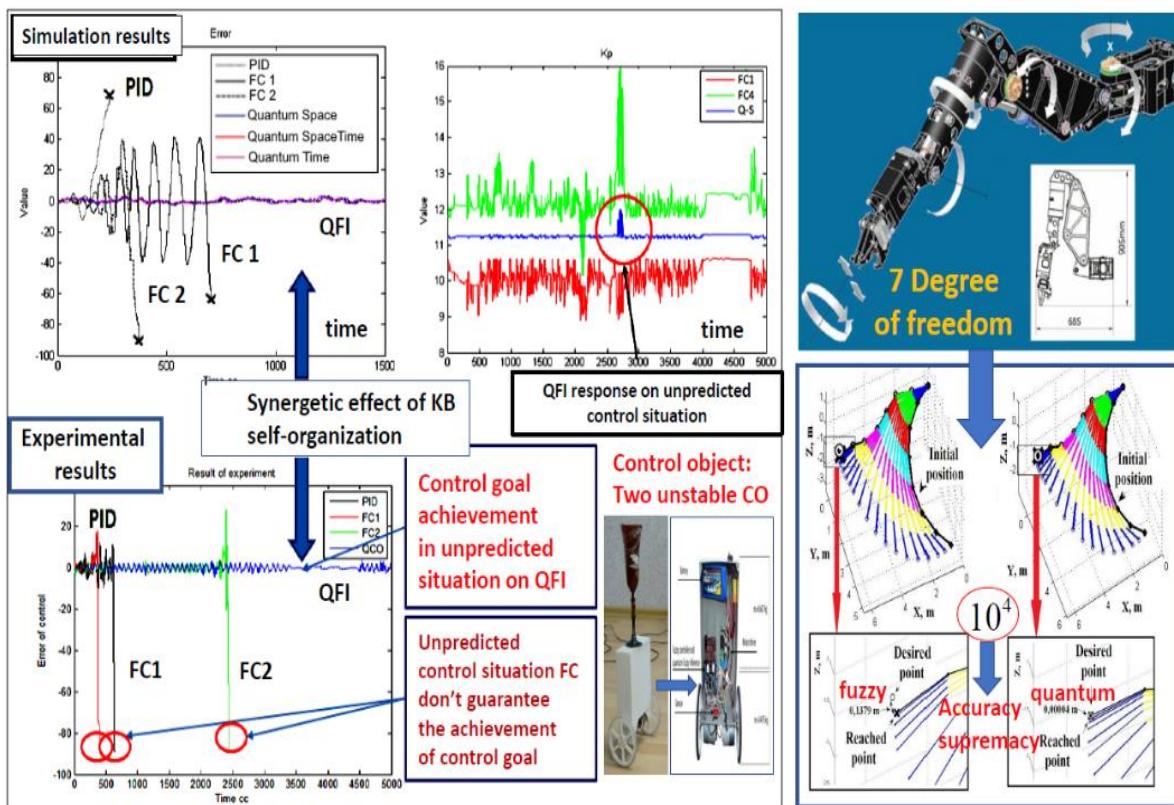


Рис. 23. Результаты моделирования и эксперимента, проведенного в непредвиденной ситуации управления для классического ПИД регулятора, нечеткого регулятором и регулятора на КНВ

На рис. 23 показаны результаты эксперимента по управлению в непредвиденных ситуациях для объекта «тележка-двойной маятник» и «избыточного манипулятора» с 7-ю степенями свободы. В эксперименте сравниваются различные контроллеры: ПИД-контроллер, два нечетких контроллера (FC_1 , FC_2) и три QFI -контроллера, основанные на различных типах корреляций: Квант-Время ($Q-T$), Квант-Пространство ($Q-S$), Квант-Пространство-Время ($Q-ST$).

Примечание. Напомним, что специфическая особенность объекта управления «каретка – двойной перевернутый маятник» (рис. 23 (а)) заключается в том, что стеклянный бокал наполнен жидкостью и не закреплен жестко на платформе автономного робота, а его математическая модель неизвестна (рассматривается как непредсказуемая ситуация управления). Система имеет две степени свободы, оба объекта управления динамически глобально неустойчивы по обобщенным координатам с существенно нелинейными связями, а в новой ситуации управления для интеллектуального робастного управления применяется только один ПИД-регулятор.

В ходе моделирования и эксперимента была применена структура робастной ИСУ, основанная на КНВ и КГА. На основе обучающего сигнала, полученного непосредственно от объекта управления, с помощью программного инструментария $QCOptKB^{\text{TM}}$ была разработана БЗ нечеткого регулятора. Нештатная ситуация моделировалась как трехкратная задержка сигнала датчика обратной связи.

Экспериментальные результаты показывают, что точность квантового регулятора, например, для избыточного манипулятора, более чем в 10 000 раз выше, чем у регулятора, основанного на мягких вычислениях. В условиях неопределенности регулятор, основанный на мягких вычислениях, резко увеличивает ошибку управления, тем самым, не достигая цели управления (см. табл. 1).

Сравнение регуляторов показывает наличие синергетического эффекта самоорганизации при проектировании робастных БЗ на основе несовершенных БЗ нечеткого регулятора. Управляющие коэффициенты ПИД-регулятора строятся на основе обратной связи несовершенной БЗ (см. «блок QFI » на рис. 1), формируя управляющее воздействие в режиме реального времени.

Табл.1. Сравнение разных контроллеров

Время, сек	Движение каретки, см					
	PID	FC1	FC2	QFI (Q-S)	QFI (Q-ST)	QFI (Q-T)
1	-1	-1	-1	1	-1	-1
2	5	3	5	5	3	4
3	-35	-4	-26	-4	-2	-3
4	60	5	36	6	4	5
5	-	-5	-60	-5	-4	-7
6	-	10	-	5	8	6
7	-	-14	-	-4	-6	-9
8	-	23	-	4	5	7
9	-	-32	-	-6	-8	-3
10	-	50	-	9	6	4
11	-	-	-	-9	-4	-7

Это достигается путем извлечения с помощью *QFI* дополнительного информационного ресурса в виде квантовой информации, скрытой в классических состояниях управляющего воздействия в виде новой ошибки управления выходным сигналом несовершенной БЗ [8,11].

Примечание. Как отмечалось ранее, в [10, 21] был предложен сокращенный квантовый генетический алгоритм (*RQGA*), который представляет собой реализацию генетического алгоритма на квантовом компьютере. Процедура поиска желаемого решения выполняется за одну операцию. Структурно алгоритм состоит из следующих шагов:

1. инициализация суперпозиции всех возможных хромосом.
2. оценка функции пригодности оператором *F*.
3. применение алгоритма Гровера.
4. использование квантового оракула.
5. использование оператора диффузии Гровера.
6. оценка решения.

Таким образом, данный инструментарий реализует принцип самоорганизации БЗ на основе квантового подхода и осуществляет оптимизацию термодинамического критерия распределения качества управления как устойчивость, управляемость и робастность [10,11].

Поэтому разработанный квантовый алгоритм нечеткого логического вывода является новым квантовым поисковым алгоритмом со встроенным квантовым генетическим алгоритмом.

Решение задачи проектирования робастного управления классическим ОУ (глобально неустойчивым и существенно нелинейным) рассматривается в качестве примера (*Benchmark*) эффективного использования разработанной информационной технологии проектирования ИСУ.

Примечание. Положительный эффект повышения уровня робастности достигается, в частности, за счет метода извлечения дополнительной квантовой информации, скрытой (и только частично доступной) в корреляционных классических состояниях законов управления, и спроектированных только на основе классических методов технологии мягких вычислений. В настоящей работе приведено описание разработки данного метода для применения в решении проблемы формирования робастных БЗ на основе квантовых стратегий принятия решений в игровых ситуациях и неопределенности ситуаций управления.

6. Симулятор робастной ИСУ роботом – манипулятором: применение ОБЗ на квантовых вычислениях

Принципиальной особенностью в построении многозвенных манипуляционных устройств является модульность, которая обеспечивает адаптивность и реконфигурируемость динамической структуры в соответствии с решаемой задачей.

Задачей управления в применении к избыточному роботу-манипулятору является:

1. обеспечение заданной точности позиционирования устройства захвата манипулятора;
2. определение пространственной конфигурации звеньев манипулятора (инвариантность обеспечивается избыточностью числа степеней свободы) с учетом непредвиденных факторов среды.

Исходя из задач управления, к конструируемой ИСУ предъявляются требования, с одной стороны, общего качества управления (в первую очередь, обеспечения достаточной точности позиционирования), а с другой, к эффективности алгоритмов решения обратной задачи динамики. Увеличение сложности структур объектов управления и трудности прогнозирования непредвиденных ситуаций усиливают внимание к поиску решения задачи гарантированного управления в условиях риска возникновения нештатных ситуаций.

Построение ИСУ роботом манипулятором возможно с применением технологий мягких вычислений. В [1, 27, 28] рассмотрены возможности проектирования ИСУ роботом манипулятором с тремя степенями свободы с единой БЗ, а также метод декомпозиции управления. Было продемонстрировано, что использование единой БЗ приводит к увеличению сложности создания БЗ, увеличению времени создания БЗ, повышению требований к вычислительным ресурсам процессора, на котором создаются БЗ и объему памяти системы, в которой размещается БЗ. Декомпозиция управления – разделение единой БЗ на несколько независимых БЗ – значительно упрощает процесс проектирования ИСУ и снижает требования к вычислительным ресурсам. Однако в результате рассогласования работы разделенных независимых БЗ несколько снижается качества управления.

В данном разделе для устранения отмеченного недостатка рассмотрен метод организации координационного управления с применением технологий квантовых вычислений [27].

6.1. Постановка задачи управления

Система управления с разделенным управлением на основе технологий мягких вычислений, спроектированная с помощью интеллектуального инструментария Оптимизатор Баз Знаний на основе мягких вычислений, представлена на рис. 24. Используется способ декомпозиции управления, когда каждый НР с заложенной БЗ отвечает за управление одним звеном (способы декомпозиции управления рассмотрены в [1]).

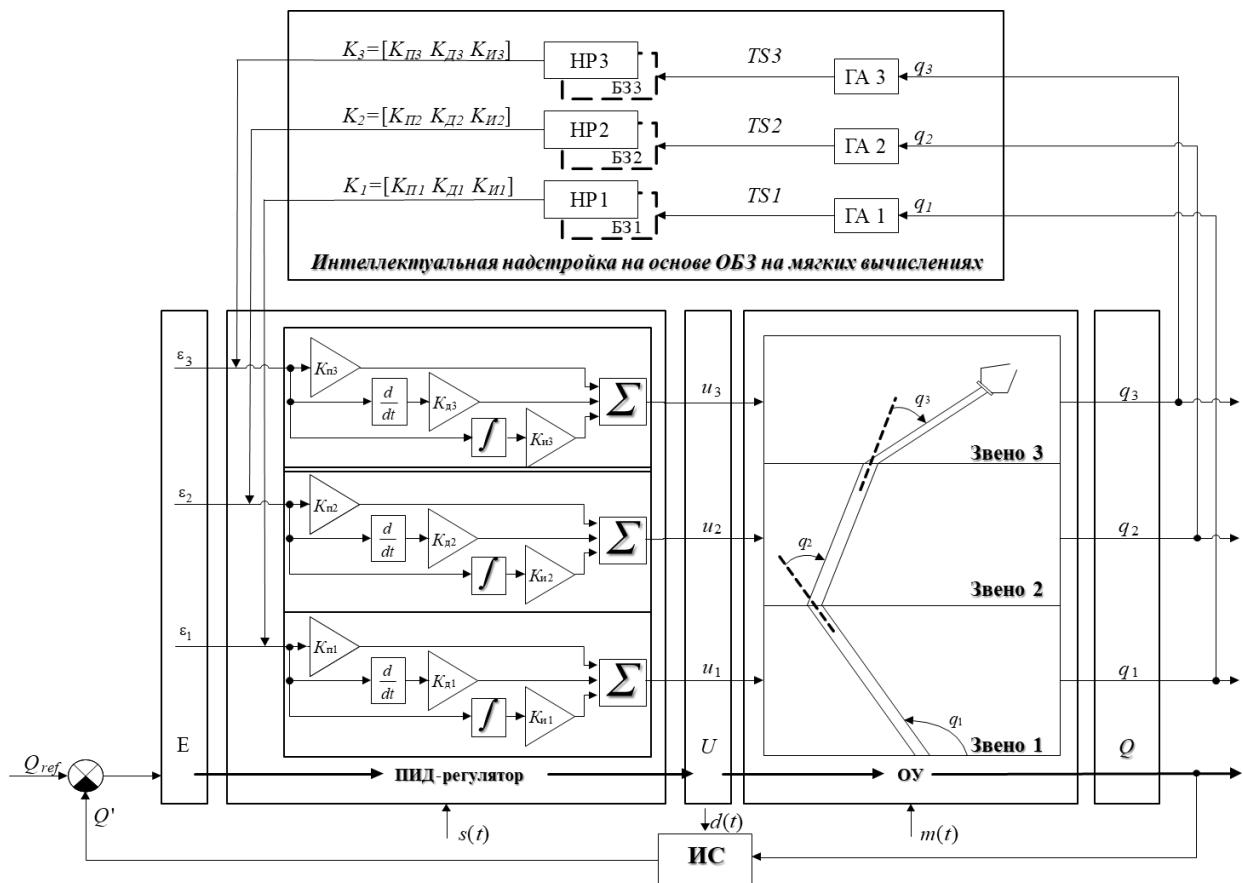


Рис. 24. ИСУ на основе ОБЗ на мягких вычислениях

На рис. 24 обозначено: Q_{ref} – воздействующий (задающий) сигнал, Q' – измеренная регулируемая величина, $E = [\varepsilon_1 \quad \varepsilon_2 \quad \varepsilon_3]$ – ошибка управления (позиционирования трех звеньев), K_{Pi}, K_{Di}, K_{Ii} , $i=1,3$ – пропорциональные, дифференциальные и интегральные коэффициенты ПИД-регулятора, i – номер соответствующего звена робота манипулятора, $s(t)$ – ограничение управляющего воздействия, $U = [u_1 \quad u_2 \quad u_3]$ – управляющее воздействие, $d(t)$ – задержка в ИС, TS_i , $i=1,3$ – обучающий сигнал соответствующего НР, $Q = [q_1 \quad q_2 \quad q_3]$ – регулируемая величина (положение соответствующего звена), $m(t)$ – внешнее воздействие среды.

Из рис. 24 видно, что задачей интеллектуальной надстройки сводится к нахождению коэффициентов ПИД-регулятора K_{Pi}, K_{Di}, K_{Ii} , $i=1,3$, при которых обеспечивается желаемый характер движения. Однако в представленной структуре коэффициенты ПИД-регулятора для каждого звена определяются независимо друг от друга, вследствие чего возникает рассогласование управления.

Для организации координационного управления в ИСУ на основе мягких вычислений с разделенным управлением, можно использовать извлечение скрытой информации о взаимосвязях существующих НР для трех звеньев манипулятора с БЗ, полученными для штатных ситуаций управления. Для этого в блок интеллектуальной надстройки необходимо включить обобщающее звено.

Мы используем для этого модель КНВ, основанную на физических законах теории квантовых вычислений [29,30] и использующую четыре оператора: суперпозицию, квантовую корреляцию, интерференцию и измерение. Первые три являются унитарными, обратимыми квантовыми операторами, а четвертый (оператор измерения) – классическим (необратимым).

Рассмотрим механизмы подключения КНВ к системе нечетких регуляторов для проектирования обобщенных сигналов управления.

6.2. Проектирование интеллектуальной системы управления с применением квантового нечеткого вывода (КНВ)

Схема извлечения скрытой информации о взаимосвязях существующих НР для трех звеньев манипулятора с БЗ, полученных для штатных ситуаций управления с использованием блока КНВ, приведена на рис. 25.

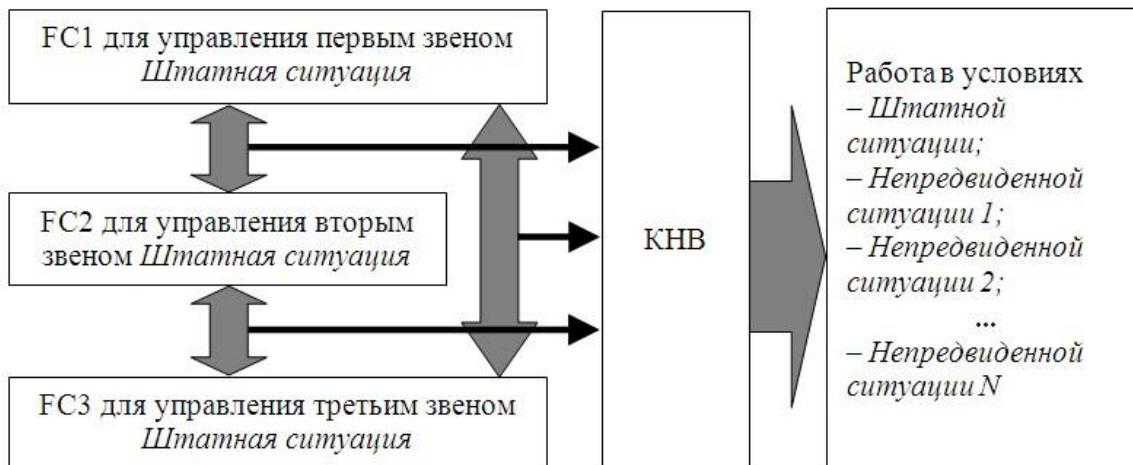


Рис. 25. Методология извлечение скрытой информации взаимосвязей БЗ НР, сконструированных для штатных ситуаций управления

Для проектирования блока КНВ использовался интеллектуальный инструментарий – Оптимизатор баз знаний на квантовых вычислениях – *QCOptKB™*.

Схема подключения блока КНВ в ИСУ представлена на рис. 26, на котором выделены блоки, спроектированные с использованием технологий мягких и квантовых вычислений. Далее ИСУ с использованием данной интеллектуальной надстройки будем называть ИСУ на основе ОБЗ на квантовых вычислениях.



Рис. 26. Интеллектуальная надстройка структуры ИСУ на основе ОБЗ на квантовых и мягких вычислениях

6.3. Симулятор квантового нечеткого вывода на основе оптимизатора баз знаний на квантовых вычислениях – *QCOptKB™*

Проектирование КНВ включает описанную ниже последовательность действий.

Шаг 1. Создание новой модели КНВ.

Для создания новой модели КНВ во вкладке «File» выбрать «New». Для создания новой модели необходимо определить количество входных и выходных переменных (поля «Input count» и «Output count» на рис. 27), формат матрицы корреляции (поле «Correlation size»).

K_{Pi}, K_{Di}, K_{Ui} , $i = 1, 3$, т.е. девяти.

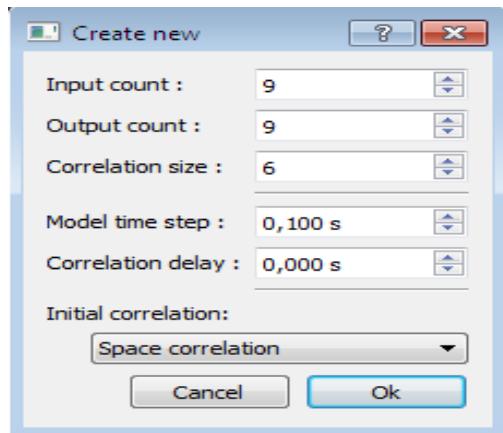


Рис. 27. Определение количества входных и выходных переменных, формата матрицы корреляции

Шаг 2. Создание гистограмм распределения амплитуд вероятностей.

Для дальнейшего удобства работы предварительно переименуем входные и выходные переменные. Для этого во вкладке «General» соответствующие поля «Input1», «Input2», «Output1», «Output2» и т.д. изменить, например, как показано на рис. 28.

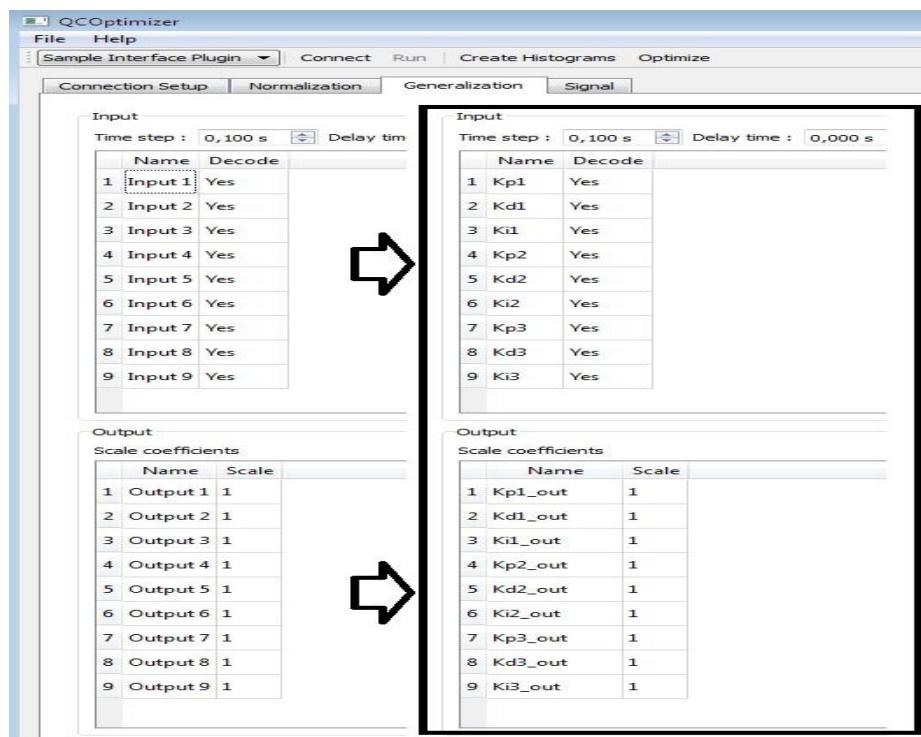


Рис. 28. Обозначение входных и выходных переменных

Для создания гистограмм распределения амплитуд вероятностей необходимо использовать сигналы на выходах НР, полученные (сохраненные как *.pat или *.mat файлы).

Например, имеем три сигнала с трех НР в виде таблиц (Таблица 2), где $\varepsilon_{Pi}, \varepsilon_{Di}, \varepsilon_{Ui}$, $i = 1, 3$ – пропорциональные, дифференциальные и интегральные ошибки, K_{Pi}, K_{Di}, K_{Ui} , $i = 1, 3$ – пропорциональные, дифференциальные и интегральные коэффициенты ПИД-регулятора, i – номер соответствующего звена робота манипулятора.

Табл. 2. Сигнал на выходе НР

ε_{Pi}	ε_{Di}	ε_{Ui}	K_{Pi}	K_{Di}	K_{Ui}
--------------------	--------------------	--------------------	----------	----------	----------

-95	-4750	-1.9	210.758	273.137	874.418
-91	-4550	-3.72	346.685	586.245	126.574
-88	-4400	-6.48	18.387	323.941	628.321
-87	-4350	-1.74	574.762	501.076	987.793
-85	-4250	-7.18	272.038	66.453	851.560
-83	-4150	-3.4	451.011	74.449	443.122
-82	-4100	-8.82	74.128	326.635	653.59
-80	-4000	-5	961.074	700.008	339.223

Для создания гистограмм необходимо выбрать меню «Create Histograms». В окне «Populate from file» выбрать файл с данными («Select data file»). После нажатия кнопки «Load» появится информация о количестве строк и столбцов в загруженном файле. В данном примере (рис. 29) подгружен файл, содержащий из 168 строк и 6 столбцов (соответственно табл. 2).

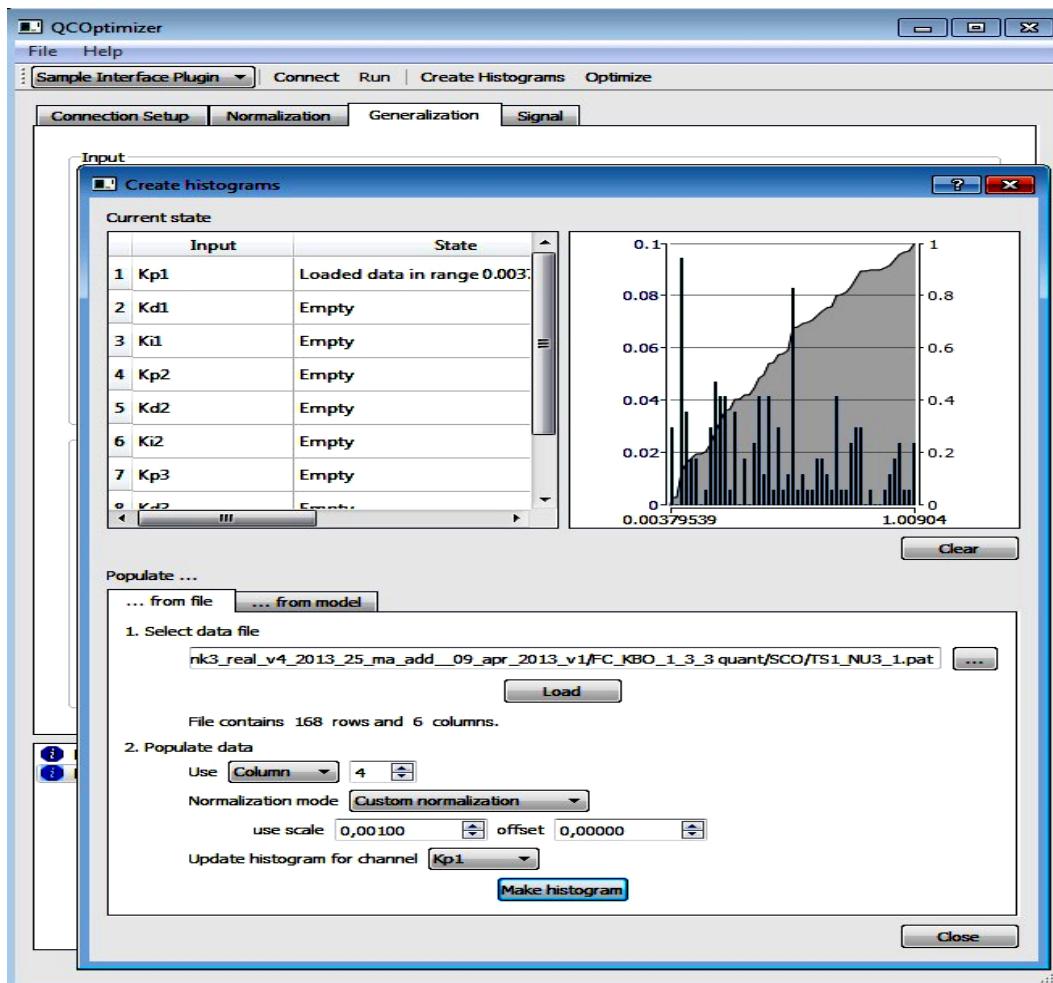


Рис. 29. Создание гистограмм распределения для входных переменных

Далее необходимо выбрать входную переменную, для которой создается гистограмма (поле «Update histogram for channel») и выбрать столбец или строку с данными (поле «Use Column/Row»). В нашем примере выбираем входную переменную K_{PI} и соответствующий ей четвертый столбец.

Следует обратить внимание, что входные сигналы должны быть нормализованными, что можно сделать, задав масштабный коэффициент (поле «use scale») и смещение (поле «offset»). Для рассматриваемого примера, коэффициенты ПИД-регулятора изменяются в диапазоне от 0 до 1000, следовательно, используем масштабный коэффициент, равный 0,001.

После нажатия кнопки «Make histogram» получаем гистограмму распределения для первой входной переменной КНВ K_{PI} .

Аналогичным образом создаются гистограммы распределения амплитуд вероятностей для других входных переменных.

Шаг 3. Определение матрицы корреляции.

Во вкладке «Generalization» в поле «Correlation matrix» (рис. 30) приведен первоначально предложенный вид матрицы корреляции.

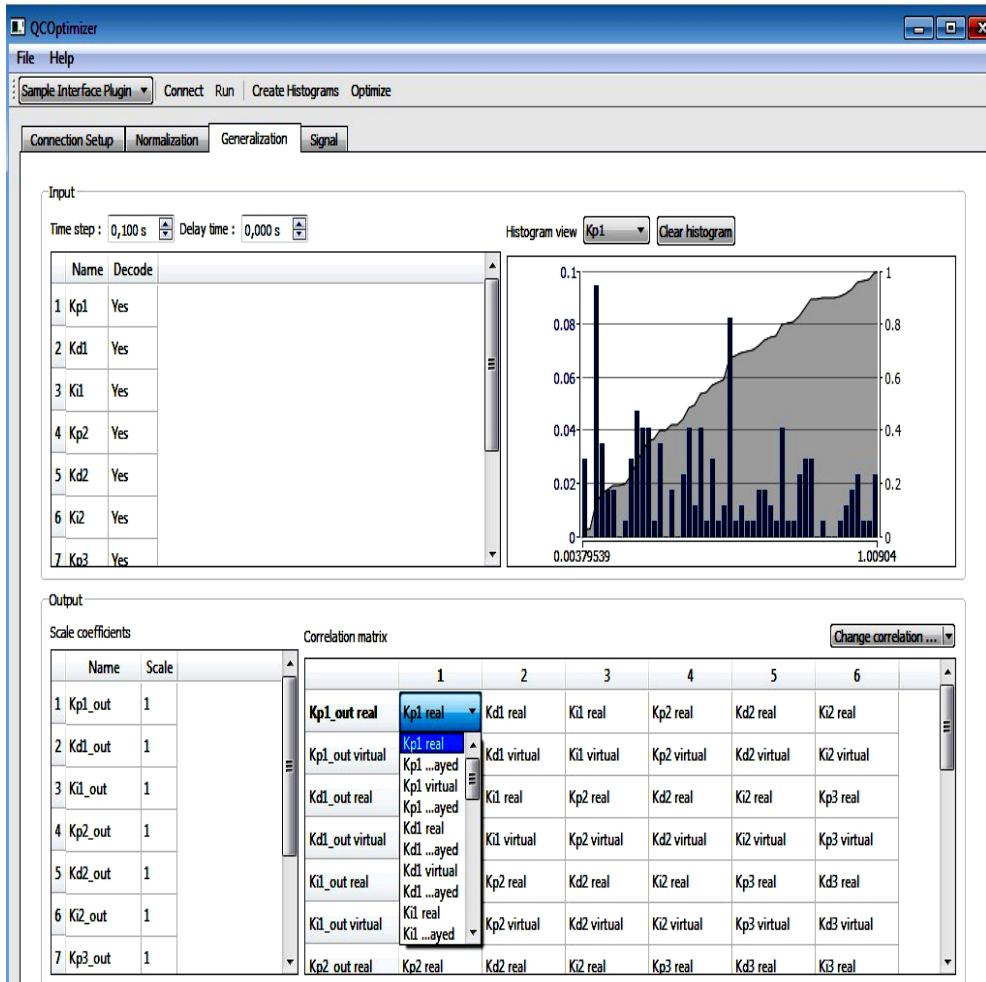


Рис. 30. Определение матрицы корреляции

Количество строк матрицы корреляции равно удвоенному количеству входных переменных (т.к. матрица корреляции содержит реальные и виртуальные состояния), количество столбцов определяется пользователем при создании новой модели КНВ корреляции (поле «Correlation size» на рис. 27).

Выбор типа квантовой корреляции (пространственной, пространственно-временной или временной) осуществляется в поле «Change correlation»

Кроме того, существует возможность задать матрицу корреляции поэлементно, как показано на рис. 30.

В нашем примере будем рассматривать три типа смешанных матриц корреляций.

A. Пространственная. Зависимость выходных сигналов $K_{\Pi}^{i_new}(t), K_{\Delta}^{i_new}(t), K_{\mathrm{И}}^{i_new}(t)$ определяется корреляцией следующих наборов входных коэффициентов:

$$\left\{ \begin{array}{l} \left\{ K_{\Pi 1}(t), K_{D1}(t), K_{\Pi 2}(t), K_{D2}(t), K_{\Pi 3}(t), K_{D3}(t) \right\} \rightarrow K_{\Pi 1}^{new}(t) \\ \left\{ K_{D1}(t), K_{I1}(t), K_{D2}(t), K_{I2}(t), K_{D3}(t), K_{I3}(t) \right\} \rightarrow K_{D1}^{new}(t) \\ \left\{ K_{I1}(t), K_{\Pi 1}(t), K_{I2}(t), K_{\Pi 2}(t), K_{I3}(t), K_{\Pi 3}(t) \right\} \rightarrow K_{I1}^{new}(t) \end{array} \right.$$

$$\left\{ \begin{array}{l} \left\{ K_{\Pi 2}(t), K_{D2}(t), K_{\Pi 3}(t), K_{D3}(t), K_{\Pi 1}(t), K_{D1}(t) \right\} \rightarrow K_{\Pi 2}^{new}(t) \\ \left\{ K_{D2}(t), K_{I2}(t), K_{D3}(t), K_{I3}(t), K_{D1}(t), K_{I1}(t) \right\} \rightarrow K_{D2}^{new}(t), \\ \left\{ K_{I2}(t), K_{\Pi 2}(t), K_{I3}(t), K_{\Pi 3}(t), K_{I1}(t), K_{\Pi 1}(t) \right\} \rightarrow K_{I2}^{new}(t) \end{array} \right.$$

$$\left\{ \begin{array}{l} \left\{ K_{\Pi 3}(t), K_{D3}(t), K_{\Pi 1}(t), K_{D1}(t), K_{\Pi 2}(t), K_{D2}(t) \right\} \rightarrow K_{\Pi 3}^{new}(t) \\ \left\{ K_{D3}(t), K_{I3}(t), K_{D1}(t), K_{I1}(t), K_{D2}(t), K_{I2}(t) \right\} \rightarrow K_{D3}^{new}(t) \\ \left\{ K_{I3}(t), K_{\Pi 3}(t), K_{I1}(t), K_{\Pi 1}(t), K_{I2}(t), K_{\Pi 2}(t) \right\} \rightarrow K_{I3}^{new}(t) \end{array} \right.$$

где каждый набор является запутанным состоянием:

$$|a_1 a_2 a_3 a_4 a_5 a_6\rangle = |K_{\Pi 1}(t), K_{\Pi 2}(t), K_{\Pi 3}(t), K_{D1}(t), K_{D2}(t), K_{D3}(t)\rangle.$$

Б. Пространственно-временная:

$$\left\{ \begin{array}{l} \left\{ K_{\Pi 1}(t), K_{D1}(t - \Delta t), K_{\Pi 2}(t), K_{D2}(t - \Delta t), K_{\Pi 3}(t), K_{D3}(t - \Delta t) \right\} \rightarrow K_{\Pi 1}^{new}(t) \\ \left\{ K_{D1}(t), K_{I1}(t - \Delta t), K_{D2}(t), K_{I2}(t - \Delta t), K_{D3}(t), K_{I3}(t - \Delta t) \right\} \rightarrow K_{D1}^{new}(t) \\ \left\{ K_{I1}(t), K_{\Pi 1}(t - \Delta t), K_{I2}(t), K_{\Pi 2}(t - \Delta t), K_{I3}(t), K_{\Pi 3}(t - \Delta t) \right\} \rightarrow K_{I1}^{new}(t) \end{array} \right.$$

$$\left\{ \begin{array}{l} \left\{ K_{\Pi 2}(t), K_{D2}(t - \Delta t), K_{\Pi 3}(t), K_{D3}(t - \Delta t), K_{\Pi 1}(t), K_{D1}(t - \Delta t) \right\} \rightarrow K_{\Pi 2}^{new}(t) \\ \left\{ K_{D2}(t), K_{I2}(t - \Delta t), K_{D3}(t), K_{I3}(t - \Delta t), K_{D1}(t), K_{I1}(t - \Delta t) \right\} \rightarrow K_{D2}^{new}(t). \\ \left\{ K_{I2}(t), K_{\Pi 2}(t - \Delta t), K_{I3}(t), K_{\Pi 3}(t - \Delta t), K_{I1}(t), K_{\Pi 1}(t - \Delta t) \right\} \rightarrow K_{I2}^{new}(t) \end{array} \right.$$

$$\left\{ \begin{array}{l} \left\{ K_{\Pi 3}(t), K_{D3}(t - \Delta t), K_{\Pi 1}(t), K_{D1}(t - \Delta t), K_{\Pi 2}(t), K_{D2}(t - \Delta t) \right\} \rightarrow K_{\Pi 3}^{new}(t) \\ \left\{ K_{D3}(t), K_{I3}(t - \Delta t), K_{D1}(t), K_{I1}(t - \Delta t), K_{D2}(t), K_{I2}(t - \Delta t) \right\} \rightarrow K_{D3}^{new}(t) \\ \left\{ K_{I3}(t), K_{\Pi 3}(t - \Delta t), K_{I1}(t), K_{\Pi 1}(t - \Delta t), K_{I2}(t), K_{\Pi 2}(t - \Delta t) \right\} \rightarrow K_{I3}^{new}(t) \end{array} \right.$$

В. Временная:

$$\left\{ \begin{array}{l} \left\{ K_{\Pi 1}(t), K_{\Pi 2}(t), K_{\Pi 3}(t), K_{\Pi 1}(t - \Delta t), K_{\Pi 2}(t - \Delta t), K_{\Pi 3}(t - \Delta t) \right\} \rightarrow K_{\Pi 1}^{new}(t) \\ \left\{ K_{D1}(t), K_{D2}(t), K_{D3}(t), K_{D1}(t - \Delta t), K_{D2}(t - \Delta t), K_{D3}(t - \Delta t) \right\} \rightarrow K_{D1}^{new}(t) \\ \left\{ K_{I1}(t), K_{I2}(t), K_{I3}(t), K_{I1}(t - \Delta t), K_{I2}(t - \Delta t), K_{I3}(t - \Delta t) \right\} \rightarrow K_{I1}^{new}(t) \end{array} \right.$$

$$\left\{ \begin{array}{l} \left\{ K_{\Pi 2}(t), K_{\Pi 3}(t), K_{\Pi 1}(t), K_{\Pi 2}(t - \Delta t), K_{\Pi 3}(t - \Delta t), K_{\Pi 1}(t - \Delta t) \right\} \rightarrow K_{\Pi 2}^{new}(t) \\ \left\{ K_{D2}(t), K_{D3}(t), K_{D1}(t), K_{D2}(t - \Delta t), K_{D3}(t - \Delta t), K_{D1}(t - \Delta t) \right\} \rightarrow K_{D2}^{new}(t) \\ \left\{ K_{I2}(t), K_{I3}(t), K_{I1}(t), K_{I2}(t - \Delta t), K_{I3}(t - \Delta t), K_{I1}(t - \Delta t) \right\} \rightarrow K_{I2}^{new}(t) \end{array} \right.$$

$$\left\{ \begin{array}{l} \left\{ K_{\Pi 3}(t), K_{\Pi 1}(t), K_{\Pi 2}(t), K_{\Pi 3}(t - \Delta t), K_{\Pi 1}(t - \Delta t), K_{\Pi 2}(t - \Delta t) \right\} \rightarrow K_{\Pi 3}^{new}(t) \\ \left\{ K_{D3}(t), K_{D1}(t), K_{D2}(t), K_{D3}(t - \Delta t), K_{D1}(t - \Delta t), K_{D2}(t - \Delta t) \right\} \rightarrow K_{D3}^{new}(t) \\ \left\{ K_{I3}(t), K_{I1}(t), K_{I2}(t), K_{I3}(t - \Delta t), K_{I1}(t - \Delta t), K_{I2}(t - \Delta t) \right\} \rightarrow K_{I3}^{new}(t) \end{array} \right.$$

Шаг 4. Установка параметров нормализации входных переменных и денормализации выходных переменных.

Блок КНВ оперирует нормализованными значениями, для чего во вкладке «Normalization» (рис. 31) можно задать масштабные коэффициенты («Scale»), смещение («Offset») или максимальные и минимальные значения (соответственно «Signal Max.» и «Signal Min.») входных переменных.

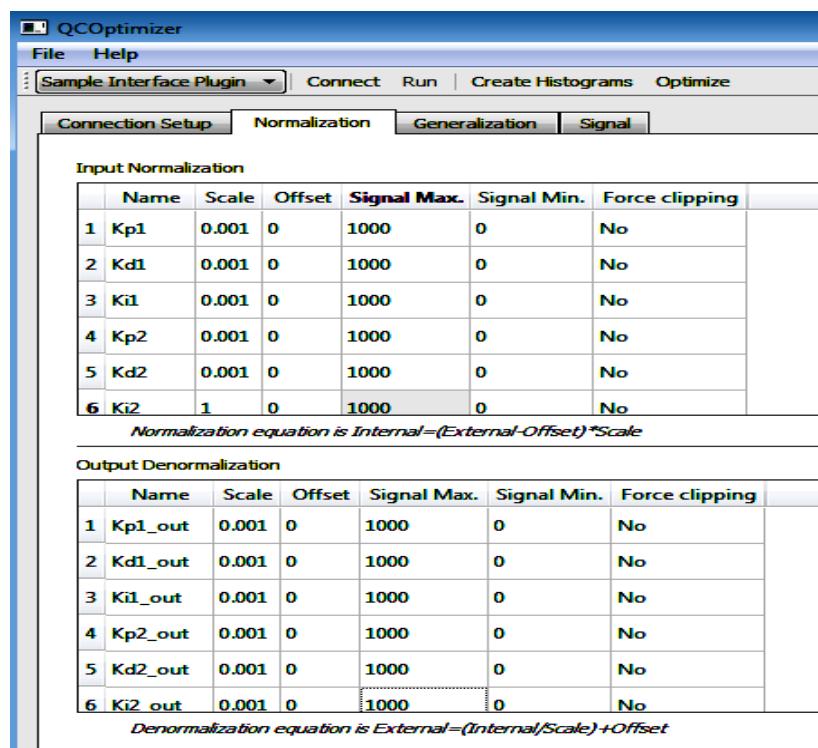


Рис. 31. Установка параметров нормализации входных переменных и денормализации выходных переменных

В нашем примере коэффициенты ПИД регулятора изменяются от 0 до 1000. Для установки параметров нормализации и денормализации достаточно для всех входных и выходных переменных в полях «Signal Max.» задать значения, равные 1000. Обратите внимание, что значения полей «Scale» автоматически установились равными 0,001.

6.4. Сравнение работы интеллектуальных систем управления с применением технологий квантовых и мягких вычислений

Сравним результаты моделирования работы ИСУ на ОБЗ на квантовых вычислениях с ИСУ на ОБЗ на мягких вычислениях с разделенным управлением. Также рассмотрим результаты физического эксперимента.

Создание лингвистических переменных для входных значений (ошибки управления, скорость ошибки управления, интегральная ошибка управления) НР. С помощью генетического алгоритма на основе ОС определяется оптимальное число функций принадлежности для каждой входной лингвистической переменной, а также выбирается оптимальная форма представления их функций принадлежности (треугольная, Гауссовская и т.д.). В качестве критерия оптимальности той или иной конфигурации лингвистических переменных используется максимум совместной информационной энтропии и минимум информации о сигналах в отдельности.

На рис. 32 показан пример построения лингвистических переменных с помощью ОБЗ.

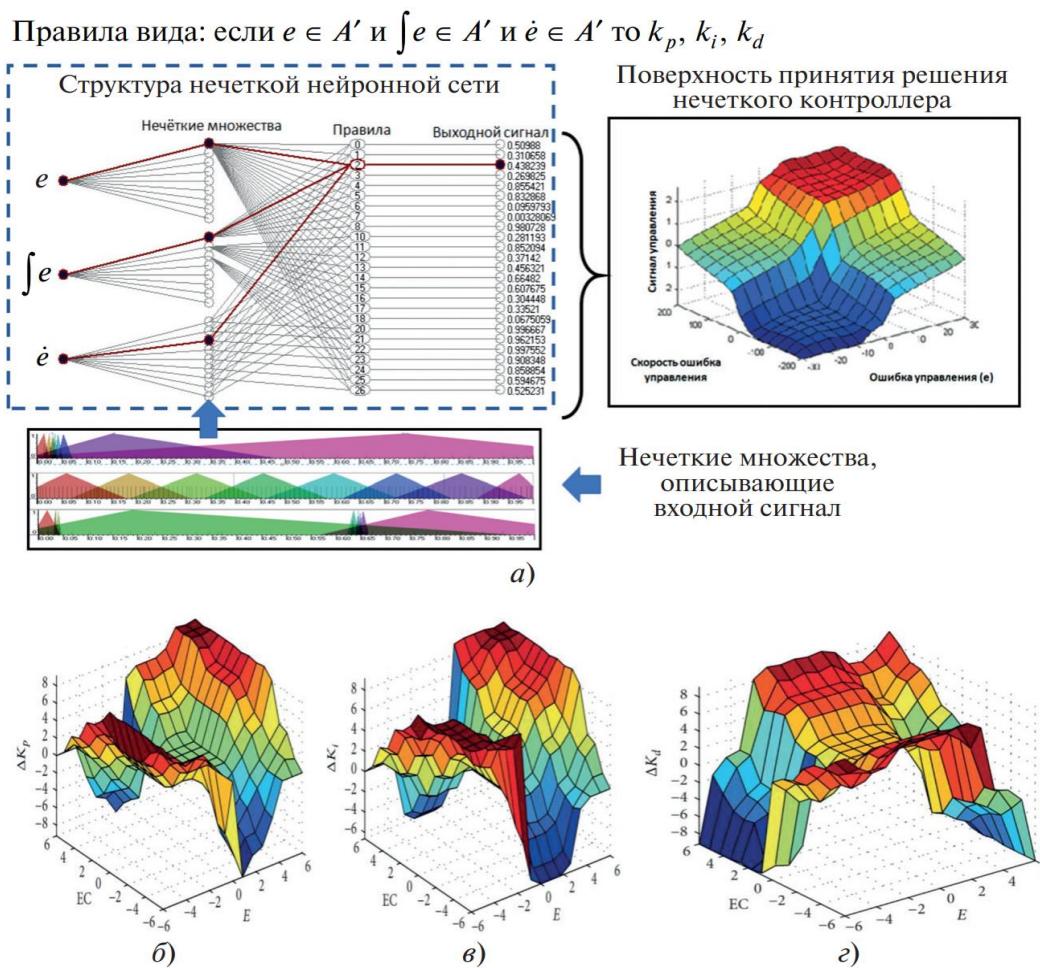


Рис. 32. Пример построения лингвистических переменных с помощью ОБЗ

Структура базы знаний в виде структуры нечеткой нейронной сети, вид функций принадлежности и поверхности управляющего нечеткого вывода НР (а); функции принадлежности для КР (б); функции принадлежности для КИ (в); функции принадлежности для КД (г). Как видно из рисунка, для описания входных значений ошибки управления ОБЗ с помощью ГА выбрал восемь функций принадлежности.

Результаты проведения серии экспериментов, для ИСУ на ОБЗ на мягких вычислениях и ИСУ на ОБЗ на квантовых вычислениях с использованием пространственной, пространственно-временной и временной корреляций, приведены соответственно в таблицах 3 и 4 для *MatLab/Simulink* моделей и

для физического макета робота манипулятора, а также на рис. 33 и 34 соответственно для *MatLab/Simulink* моделей и для физического макета робота манипулятора.

Табл. 3. Сравнение критериев качества для *MatLab/Simulink* моделей

	Вид решения	ИСУ на ОБЗ на мягких вычислениях с разделенным управлением	ИСУ на ОБЗ на квантовых вычислениях		
			Использование пространственной корреляции	Использование пространственно-временной корреляции	Использование временной корреляции
1	Решение в штатных ситуациях	1	1	1	1
2	Решение в непредвиденных ситуациях	0,833333	1	1	1
3	Быстродействие	0,516917	0,807667	0,8135	0,743833
4	Относительное значение перерегулирования	0,996353	1	1	0,999155
5	Относительная ошибка позиционирования	0,888263	0,99815	0,99815	0,996413
3	Время одной итерации	0,96783	0,922353	0,923098	0,93069
7	Сложность реализации управления	0,948788	0,98847	0,985205	0,986955
8	Общая оценка управления	0,819583	0,953007	0,954085	0,941382

Табл. 4. Сравнение критериев качества для физического макета робота манипулятора

	Вид решения	ИСУ на ОБЗ на мягких вычислениях с разделенным управлением	ИСУ на ОБЗ на квантовых вычислениях		
			Использование пространственной корреляции	Использование пространственно-временной корреляции	Использование временной корреляции
1	Решение в штатных ситуациях	1	1	1	1
2	Решение в непредвиденных ситуациях	0,9	1	1	1
3	Быстродействие	0,569137	0,779917	0,602137	0,716833
4	Относительное значение перерегулирования	0,9955	0,998225	0,998075	0,997425
5	Относительная ошибка позиционирования	0,989375	0,998225	0,99475	0,996533
3	Время одной итерации	0,75755	0,72541	0,748325	0,73105
7	Сложность реализации управления	0,935675	0,97315	0,976525	0,998
8	Общая оценка управления	0,845085	0,926916	0,893471	0,915896

Оценка работы систем управления производится по серии экспериментов с использованием введенной системы критериев качества (описание экспериментального макета, методики проведения экспериментов и критериев качества приведено в [27, 28]).

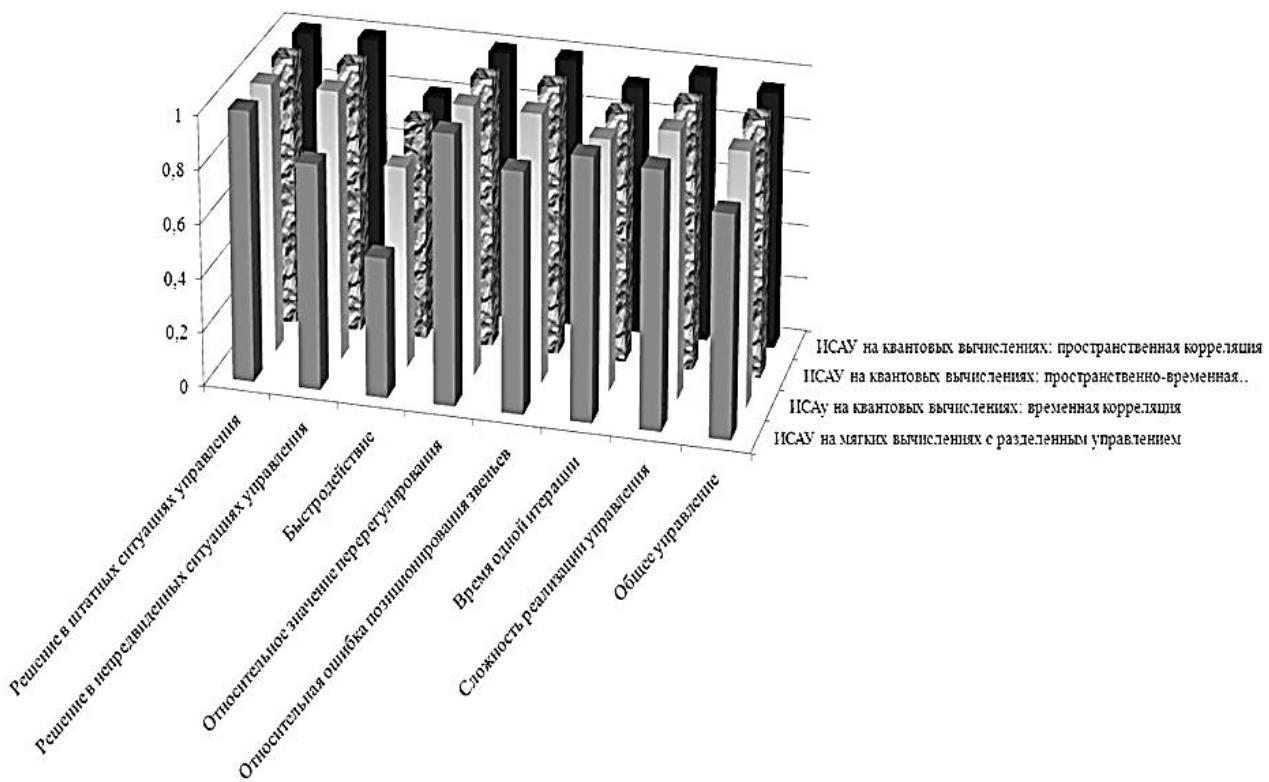


Рис. 33. Сравнение результатов работы ИСУ на ОБЗ на мягких вычислениях с разделенным управлением, ИСУ на ОБЗ на квантовых вычислениях с применением пространственной, пространственно-временной и временной корреляций при тестировании MatLab/Simulink моделей

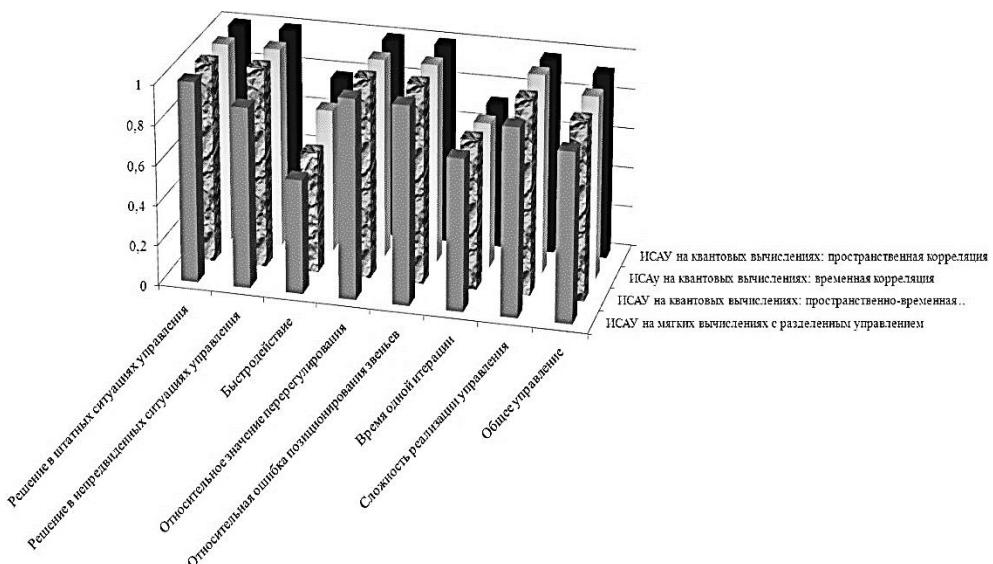


Рис. 34. Сравнение результатов работы ИСУ на ОБЗ на мягких вычислениях с разделенным управлением, ИСУ на ОБЗ на квантовых вычислениях при тестировании физического макета робота манипулятора

Из таблиц 3 и 4, а также из рис. 33 и 34 видно, что при включении в структуру ИСУ с разделенным управлением блока КНВ, критерий «Решение задачи позиционирования в непредвиденных ситуациях управления» достигает максимального значения, улучшается показатель критерия «Сложность реализации управления» при использовании любого из трех рассмотренных типов корреляций как для *MatLab/Simulink* моделей, так и для физического макета робота манипулятора.

В связи с появлением в системе управления дополнительного звена КНВ, в ИСУ на ОБЗ на квантовых вычислениях по сравнению с ИСУ на ОБЗ на мягких вычислениях с разделенным управлением, ухудшается критерий «Время одной итерации».

Значительно улучшаются показатели критерия «Быстродействие».

В целом, «Общая оценка управления» улучшается во всех случаях при использовании ИСУ на ОБЗ на квантовых вычислениях по сравнению с использованием ИСУ на ОБЗ на мягких вычислениях с разделенным управлением.

Если для *MatLab/Simulink* моделей наилучший показатель обеспечивается при использовании ИСУ на ОБЗ на квантовых вычислениях с применением пространственно-временной корреляции, то физическое тестирование Модуля Манипулятора определяет оптимальным выбор пространственной корреляции.

Для наглядности, рассмотрим результаты сравнения работы *MatLab/Simulink* моделей в условиях непредвиденной ситуации управления: на 11-й итерации положение второго звена принудительно изменяется (рис. 35 – изменение положения второго звена манипулятора под управлением ИСУ на ОБЗ на мягких вычислениях с разделенным управлением).

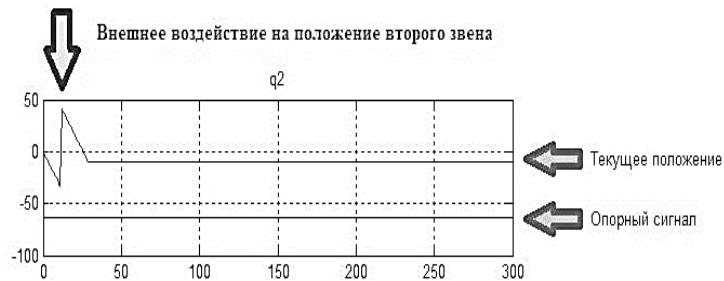


Рис. 35. Моделирование внешнего воздействия на второе звено манипулятора

Построим траектории движения манипулятора с тремя степенями свободы под управлением ИСУ на ОБЗ на мягких вычислениях с разделенным управлением и ИСУ на ОБЗ на квантовых вычислениях с применением пространственной корреляции в условиях обозначенной непредвиденной ситуации (рис. 36).

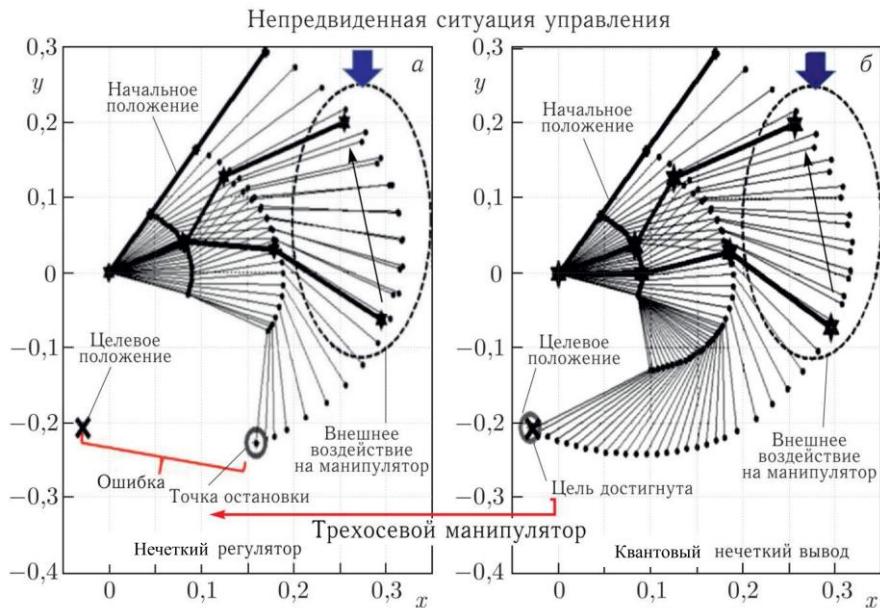


Рис. 36. Поведение манипулятора с тремя степенями свободы в непредвиденной ситуации: а) под управлением ИСУ на ОБЗ на мягких вычислениях с разделенным управлением; б) под управлением ИСУ на ОБЗ на квантовых вычислениях с применением пространственной корреляции

На рис. 37 приведена динамика изменения коэффициентов ПИД-регулятора на входе и выходе КНВ (соответственно левый и правый столбцы) для рассматриваемой нештатной ситуации.

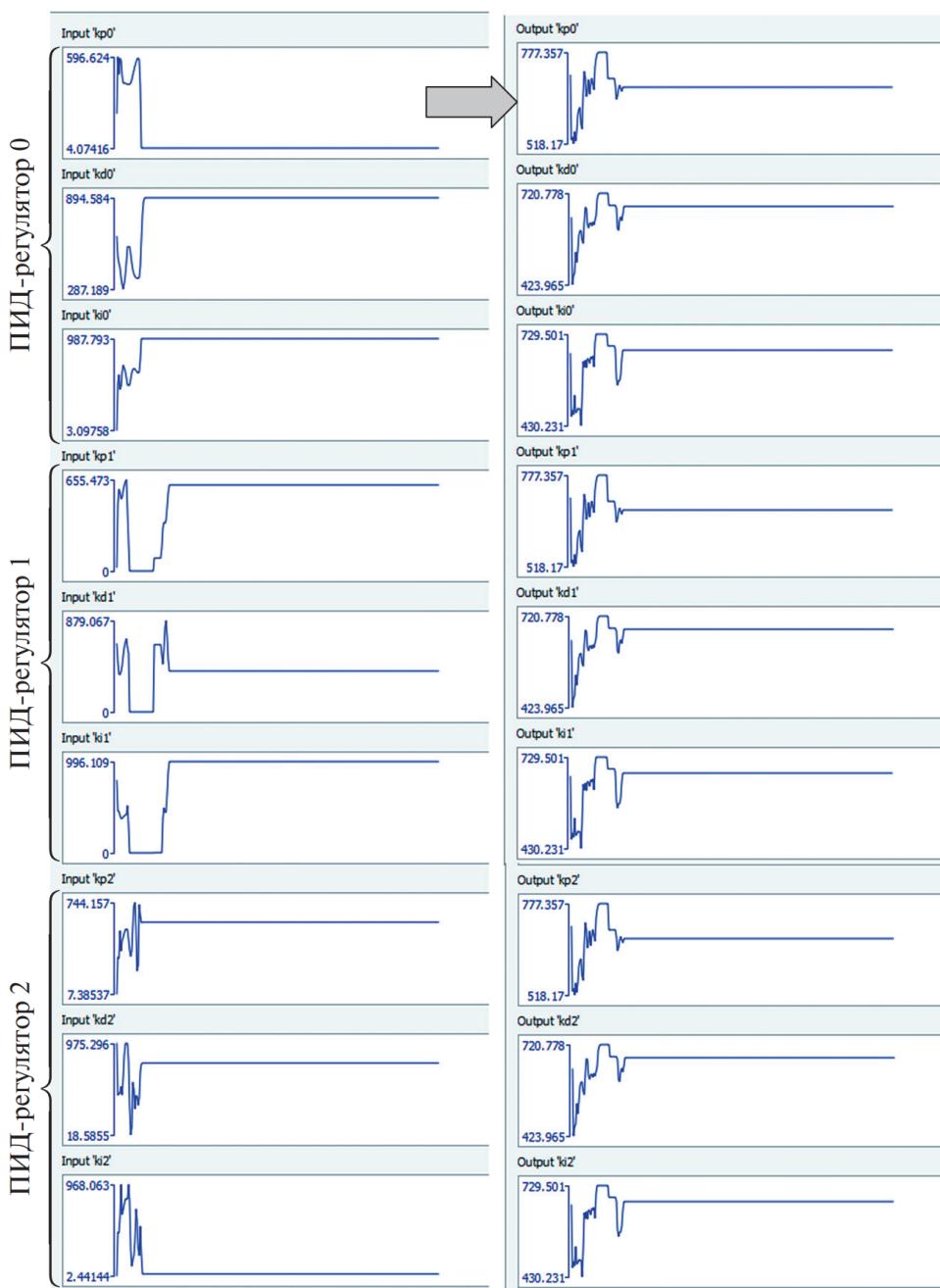


Рис. 37. Динамика изменения коэффициентов ПИД-регулятора на входе и выходе КНВ

Из рис. 37 видно, что в отличие от ИСУ на ОБЗ на мягких вычислениях с разделенным управлением, ИСУ на ОБЗ на квантовых вычислениях справляется с задачей позиционирования с заданной точностью в рассмотренной непредвиденной ситуации управления. Неспособность ИСУ на ОБЗ на мягких вычислениях решить задачу точного позиционного управления иллюстрируется также на рис. 37 – нечеткий регулятор, отвечающий за управления вторым звеном за отведенное время работы так и не смог «реабилитироваться» после мощного внешнего воздействия, в результате ошибка позиционирования второго звена составила более 50 град.

На рис. 38 представлено сравнение законов формирования сил управления (U_1, U_2, U_3) для трех звеньев манипулятора: сгенерированных с помощью ИСУ на ОБЗ на мягких вычислениях с разделенным управлением (толстые линии) и ИСУ на ОБЗ на квантовых вычислениях (тонкие линии).

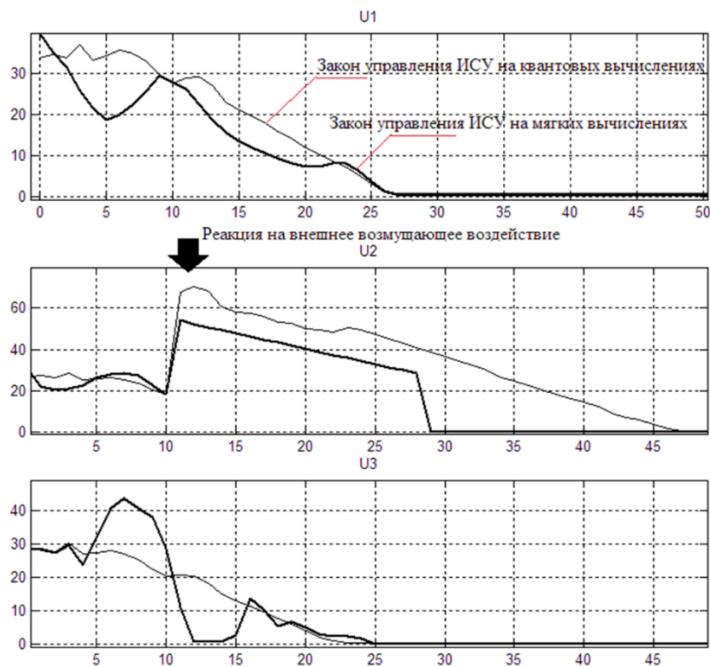


Рис. 38. Сравнение законов управления ИСУ на ОБЗ на мягких вычислениях с разделенным управлением и ИСУ на ОБЗ на квантовых вычислениях

Из рис. 38 видно, что включение в ИСУ блока КНВ дает возможность получать адекватные к возмущениям сигналы управления с меньшими плавными колебаниями, чем на выходе нечеткого регулятора.

Рассмотренные ранее случаи непредвиденных ситуаций управления – изменение начальных условий, принудительное смещение звеньев – относятся к внешним непредвиденным ситуациям управления.

Однако помимо внешних возмущений, возможны и изменения во внутренней конфигурации объекта и системы управления, связанные с неполнотой начального описания, помехами в управляющих каналах, неточностью и инерционностью измерительной системы и т.д.

Внесем в модели систем управления дополнительные случаи непредвиденных ситуаций управления, связанных с изменениями параметров ОУ во времени.

6.5. Моделирование непредвиденных ситуаций управления в условиях изменений параметров ОУ

Введем два случая непредвиденных ситуаций управления (начальные условия соответствуют штатным ситуациям управления):

- 1) на одиннадцатой итерации изменяется ограничение выходного воздействия в терминах изменений положений звеньев: увеличивается с 3 до 5 град.
- 2) на одиннадцатой итерации изменяется ограничение выходного воздействия в терминах изменений положений звеньев: уменьшается с 3 до 1 град.

Сравним законы управления, сформированные ИСУ на ОБЗ на квантовых вычислениях при использовании пространственной, пространственно-временной и временной корреляций (рис. 39) для непредвиденных ситуаций управления в условиях изменения параметров ОУ.

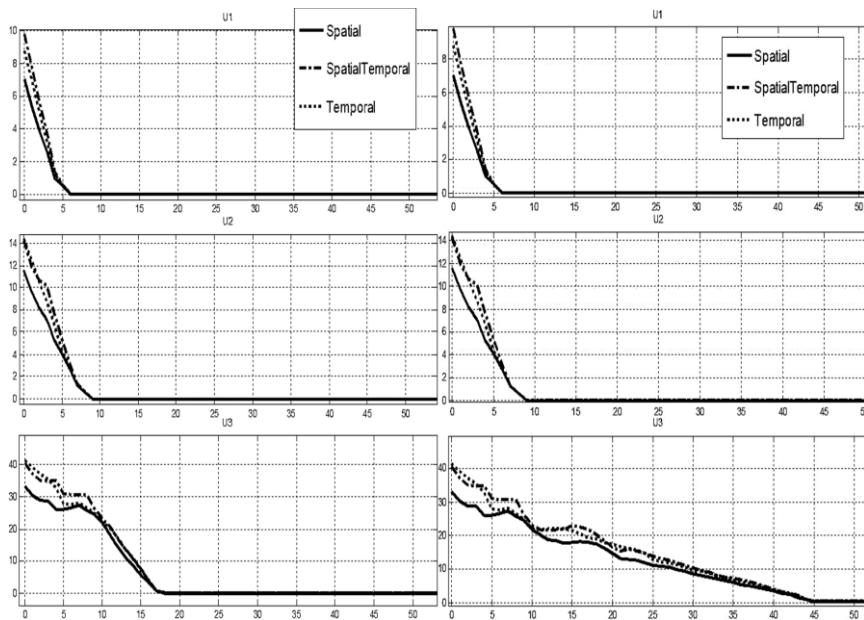


Рис. 39. Сравнение сформированных ИСУ на ОБЗ на квантовых вычислениях законов управления с применением пространственной пространственно-временной и временной корреляций в непредвиденных ситуациях управления в условиях изменения параметров ОУ: первый случай (слева); второй случай (справа)

На рис. 39 законы управления, сформированные ИСУ на ОБЗ на квантовых вычислениях с применением определенного типа корреляции обозначены следующим образом: *Spatial* – пространственная корреляция, *Spatio-Temporal* – пространственно-временная корреляция, *Temporal* – временная корреляция. Из рис. 39 видно, что при формировании управляющего воздействия минимальный расход полезного ресурса достигается при использовании пространственной корреляции.

Для рассмотренных экспериментов (непредвиденные ситуации управления в условиях изменения параметров ОУ) сравним законы управления, формируемые с помощью ИСУ на ОБЗ на квантовых вычислениях с использованием пространственной корреляции, а также ИСУ на ОБЗ на мягких вычислениях с одним НР и разделенным управлением (рис. 40).

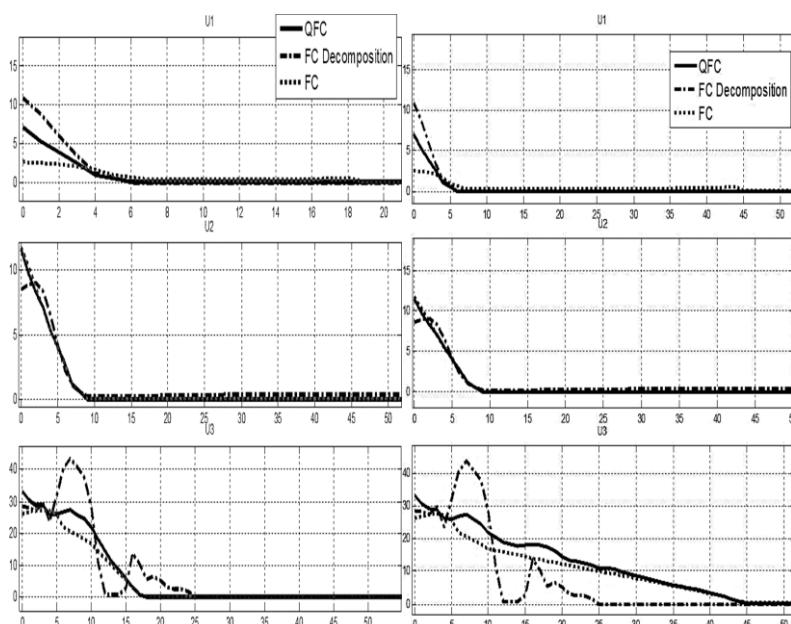


Рис. 40. Сравнение сформированных законов управления: ИСУ на ОБЗ на квантовых вычислениях с применением пространственной корреляции, ИСУ на ОБЗ на мягких вычислениях с одним НР и ИСУ

на ОБЗ на мягких вычислениях с разделенным управлением в непредвиденных ситуациях управления в условиях изменения параметров ОУ: первый случай (слева); второй случай (справа)

На рис. 40 законы управления обозначены следующим образом: *QFC* – сформированные с помощью ИСУ на ОБЗ на квантовых вычислениях с использованием пространственной корреляции, *FC Decomposition* – с помощью ИСУ на ОБЗ на мягких вычислениях с разделенным управлением, *FC* – с помощью ИСУ на ОБЗ на мягких вычислениях с одним НР.

Для рассмотренных экспериментов (рис. 40) с точки зрения минимизации расхода полезного ресурса, наилучшие законы управления формируются с помощью ИСУ на ОБЗ на мягких вычислениях с одним НР, незначительное ухудшение наблюдается при формировании с помощью ИСУ на ОБЗ на квантовых вычислениях с применением пространственной корреляции. Законы управления, сформированные ИСУ на ОБЗ с разделенным управлением в условиях изменения параметров ОУ, приобрели колебательный характер.

Рассмотрим другую пару экспериментов, также соответствующую двум рассматриваемым случаям непредвиденных ситуаций управления в условиях изменения параметров ОУ (рис. 41).

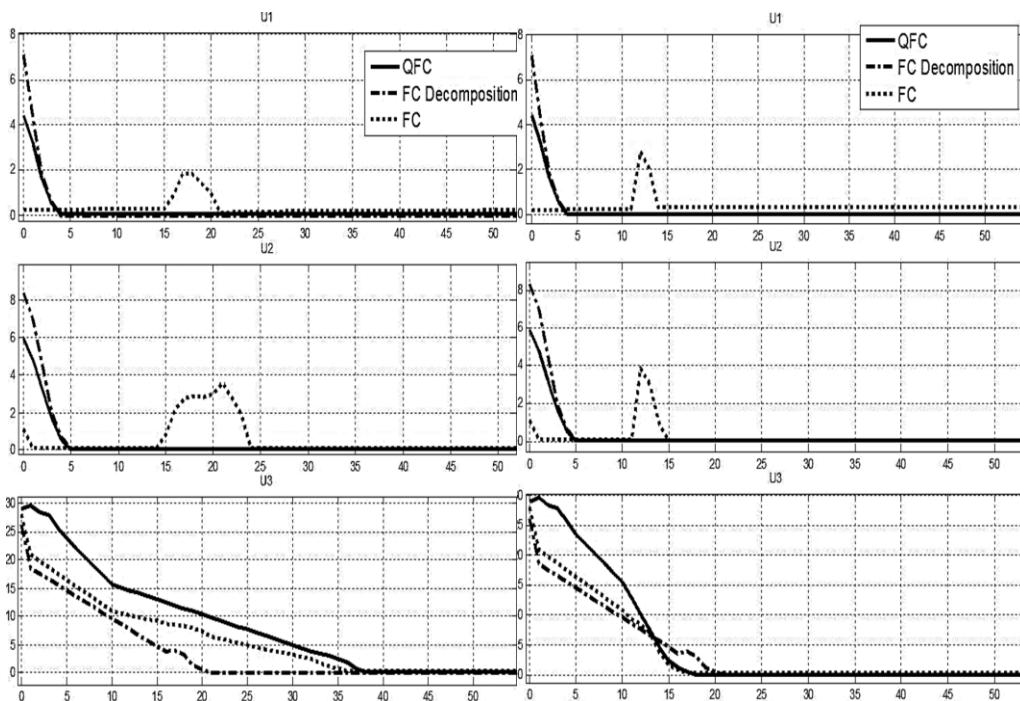


Рис. 41. Сравнение сформированных законов управления: ИСУ на ОБЗ на квантовых вычислениях с применением пространственной корреляции, ИСУ на ОБЗ на мягких вычислениях с одним НР и ИСУ на ОБЗ на мягких вычислениях с разделенным управлением в непредвиденных ситуациях управления в условиях изменения параметров ОУ: первый случай (слева); второй случай (справа)

При рассмотрении экспериментов (рис. 41), стабильные законы управления были сформированы ИСУ на ОБЗ на квантовых вычислениях с применением пространственной корреляции и ИСУ на ОБЗ на мягких вычислениях с разделенным управлением, тогда как в законах управления, сформированных ИСУ на ОБЗ на мягких вычислениях с одним НР присутствуют участки локально неустойчивых состояний.

В рассмотренных примерах с точки зрения качества формирования законов управления (рис. 39 - 41), наилучшими свойствами обладает ИСУ на ОБЗ на квантовых вычислениях с использованием пространственной корреляции, в то время как с помощью ИСУ на ОБЗ на мягких вычислениях с разделенным управлением был сформирован вектор управляющих воздействий с колебательным характером, а с помощью ИСУ на ОБЗ на мягких вычислениях – управляющие воздействия с локально неустойчивыми состояниями.

Таким образом, минимальный расход полезного ресурса при формировании управляющих сигналов обеспечен при использовании ИСУ на основе ОБЗ на квантовых вычислениях. Нечеткие поверхности параметров K_p гибридного нечеткого ПИД-регулятора для второго звена, например, до и после воздействия представлены на рис. 42.

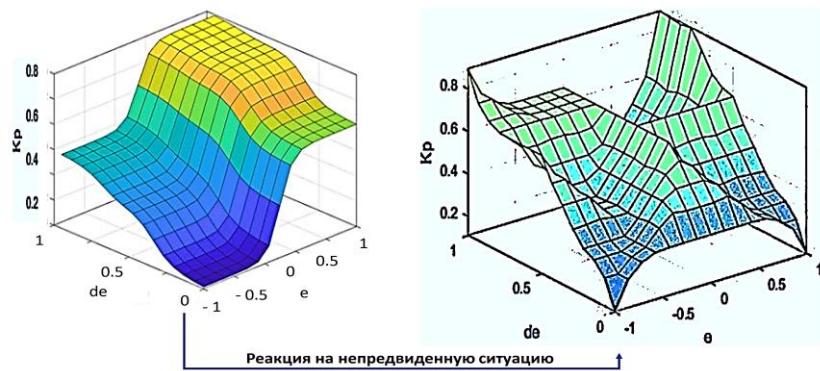


Рис. 42. Нечеткая поверхность параметров K_p гибридного нечеткого ПИД-регулятора для второго звена до (a) и после (b) воздействия

Нечеткие поверхности параметров K_p гибридного нечеткого ПИД-регулятора для третьего звена с прикрепленным на конце захватным устройством, до и после воздействия представлены на рис. 43.

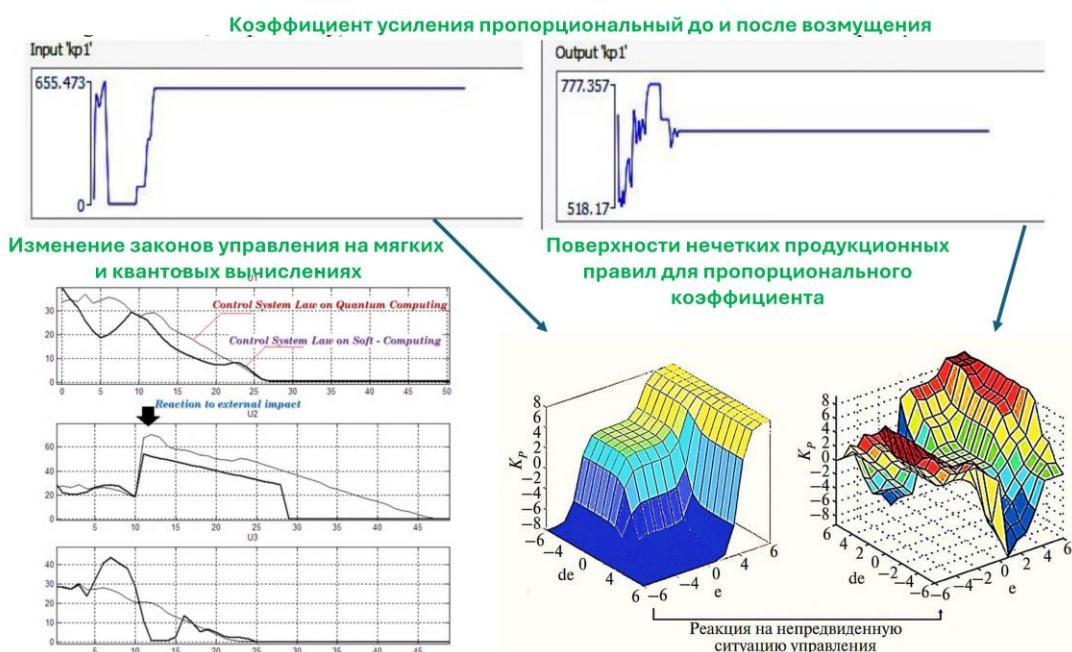


Рис. 43. Нечеткая поверхность параметров K_p гибридного нечеткого ПИД-регулятора для третьего звена до (a) и после (b) воздействия

Динамика изменения коэффициентов ПИД-регулятора на входе и выходе КНВ приведена на рис. 37.

Этот результат особенно примечателен тем, что при организации координационного управления за счет одной БЗ (соответственно использование одного нечеткого регулятора в ИСУ для ОБЗ по мягким вычислениям) количество входных переменных - т.е. параметров, определяющих функционирование системы, ограничено вычислительными ресурсами процессора, на котором создана БЗ нечеткого регулятора, и объемом памяти в системе, в которой находится нечеткий регулятор. Более того, для сложных систем, таких как робот-манипулятор с 7DoF, организация одной БЗ нечеткого регулятора невозможна.

7. Анализ результатов моделирования

В целом, возможность декомпозиции управления (разделения одной БЗ на несколько одинаковых независимых БЗ) и организации координационного управления путем введения квантованного нечеткого факторного звена значительно увеличивает возможное количество входных переменных и тем самым расширяет возможности учета параметров системы и объекта управления.

7.1. Физическая интерпретация

- Результаты моделирования показывают, что с помощью КНВ можно из двух не робастных выходов БЗ спроектировать оптимальный робастный сигнал управления с простыми «мудрыми» законами управления графиком коэффициентов усиления ПИД в непредсказуемых ситуациях управления.
- И это несмотря на то, что в непредвиденных ситуациях управления нечеткие регуляторы теряют свойство робастность.
- Физически это является демонстрацией выполнения принципа минимума энтропии относительно совершения полезной работы на основе извлеченных квантовых знаний, скрытых в классических состояниях.

Дальнейшие исследования направлены на разработку и анализ физического манипулятора с 7DOF, а также его интеграцию с мобильной платформой [5-7]. Дополнительные к изложенному результаты приведены в [13, 27, 28, 31-36].

7.2. Результаты

ОБЗ позволяет моделировать различные версии БЗ (*multiple-KB*) нечетких регуляторов, которые гарантируют робастность для фиксированных сред управления.

- Вводится принцип самоорганизации в квантовой БЗ с минимальной энтропией в интеллектуальных состояниях управления в режиме on-line.
- Блок КНВ повышает робастность нечеткого регулятора, используя способность к самоорганизации.
- Разработанная квантовая система управления достигает поставленных целей управления во многих непредсказуемых ситуациях управления: надежность ИСУ на основе КНВ повышается в непредсказуемых ситуациях управления.
- Используя ОБЗ и КНВ, можно проектировать «разумное» управление существенно нелинейными устойчивыми и, особенно, неустойчивыми динамическими системами в условиях неопределенности информации о внешних возбуждениях и изменения опорных сигналов (цели управления) и параметров модели.
- Законы управления, основанные на КНВ, просты для физической реализации.
- Нечеткий регулятор на основе КНВ требует минимум исходной информации о внешней среде и внутренней структуре модели объекта управления.
- Для извлечения ценностной информации для «разумного» управления и проектирования унифицированного робастного БЗ в квантовом нечетком регуляторе используется on-line процесс.
- КНВ, основанный на двух субоптимальных БЗ, улучшает робастность субоптимальных БЗ;
- Даже в случае субоптимальных БЗ, КНВ на основе двух субоптимальных БЗ проявляет хорошие способности к самоорганизации.
- С точки зрения квантовой теории игр имеем парадокс *«Parrondo»*: из двух классических БЗ, которые не являются победителями в различных непредвиденных условиях, с помощью инструментария КНВ мы можем создать одного победителя в качестве «мудрого» управляющего сигнала, используя квантовую стратегию принятия решений (без запутывания). Этот эффект в робастном управлении был описан также в [8,15,35,36] на других примерах неустойчивых систем управления (подробности описания технологии см. на сайте: <http://www.qcoptimizer.com/>).

Приведенные результаты позволяют более полно и глубже понять решение следующей трудной и принципиально важной для теории и систем управления проблемы: определение роли и влияния квантовых синергетических эффектов на повышение уровня робастности проектируемых интеллектуальных процессов управления.

Выводы

- Полноценная разработка и инженерное прикладное применение квантовых алгоритмов при текущих исследованиях и практических результатах в области квантовых компьютеров затруднено. Необходимо делать исследования с учётом доступных возможностей к разработке и моделировать квантовые алгоритмы на классическом компьютере, ожидая возможности создания промышленных версий квантовых компьютеров. Применение квантового программирования и супер-компьютинга на основе GRID-технологий, а также разработка квантовых симуляторов является одними из эффективных решений в данном направлении работ.
- Одной из востребованной задачей является интеллектуальное робастное управление физическими объектами в непредвиденных ситуациях. Современные решения этой проблемы уже могут дать хороший результат, но такие системы не могут обучаться в режиме реального времени, поэтому набор способов реакций на события крайне ограничен. Применение квантовых вычислений и, в частности, алгоритма КНВ позволяет решить подобные проблемы за счёт повышенной скорости глубокого машинного обучения.
- Создание симулятора для управления существенно нелинейными динамическими глобально неустойчивыми системами (в данной статье – «каретка – маятник») позволило решить целый ряд проблем при разработке алгоритмов ИСУ, а также упростить разработку собственных алгоритмов для исследователей в области интеллектуального робастного управления.
- Основное внимание уделено использованию квантового нечёткого вывода и выбора типа корреляции. Автоматизация выбора оптимального типа квантовой корреляции влияет на улучшение робастности системы. Показан способ нахождения оптимального типа квантовой корреляции – при помощи квантового генетического алгоритма. Проведено сравнение трех квантовых генетических алгоритмов и особое внимание уделено сокращённому квантовому генетическому алгоритму с использованием алгоритма Гровера и квантового оракула. Подобный алгоритм позволяет проводить отбор в одном поколении. Однако применение его на классическом компьютере возможно только в простых задачах, а на большом количестве возможных решений алгоритм сейчас слабо эффективен.
- Выбранный квантовый генетический алгоритм был разработан для поиска оптимального типа квантовой корреляции и проведен анализ его работы на примере системы «каретка-перевёрнутый маятник», где тип корреляции, выбранный квантовым генетическим алгоритмом, совпал с типом корреляции, показывающий лучшие результаты при ручном выборе.

В результате сравнения работы рассмотренных типов систем управления роботом манипулятором с тремя степенями свободы, можно сделать следующие выводы:

- общая оценка качества управления выше в случае применения ИСУ на ОБЗ на квантовых вычислениях (для пространственной, пространственно-временной и временной корреляций) по сравнению с ИСУ на ОБЗ на мягких вычислениях с разделенным управлением, что является следствием введения в структуру ИСУ дополнительного звена КНВ, организующего координационное управление;
- в результате тестирования *MatLab/Simulink* моделей лучшими показателями обладала ИСУ, использующая пространственно-временную корреляцию; однако тестирование физического макета робота манипулятора определило наиболее оптимальным использование пространственной корреляции;
- минимальный расход полезного ресурса достигается при использовании в ИСУ на ОБЗ на квантовых вычислениях пространственной корреляции по сравнению с пространственно-временной и временной корреляциями;
- законы управления, сформированные ИСУ на ОБЗ на мягких вычислениях с разделенным управлением в условиях изменения параметров ОУ или внешних возмущающих воздействий,

приобрели колебательный характер, а в законах управления, сформированных ИСУ на ОБЗ с одним НР появились участки локально неустойчивых состояний.

Возможность декомпозиции управления (разделения одной БЗ на несколько однотипных независимых БЗ) и организация координационного управления с помощью введения звена КНВ, существенно увеличивает возможное число входных переменных и тем самым расширяет возможности учета параметров системы и объекта управления.

- В данной статье продемонстрированы основные идеи квантовых вычислений и квантовой теории информации применительно к разработке стратегии и методов проектирования надежной базы знаний с использованием квантового нечеткого вывода. Квантовый нечеткий вывод может быть эффективно реализован на персональном компьютере стандартной конфигурации.
- Показана необходимость применения нового типа вычислительного интеллекта (*quantum soft computing*) для эффективного решения классических задач управления, неразрешимых алгоритмически, таких как робастность управления. Приведено описание программного инструментария процессов проектирования робастных баз знаний для интеллектуальных систем управления на основе квантовых нечетких вычислений.
- Робастность интеллектуальных нечетких регуляторов была продемонстрирована на примере самоорганизующейся интеллектуальной системы управления локально неустойчивым и существенно нелинейным управляемым объектом. На основе результатов моделирования законов управления коэффициентами усиления нечетких ПИД-регуляторов, полученных с помощью квантовой модели нечеткого вывода, можно сформулировать новый принцип проектирования робастных структур интеллектуальных систем управления: проектирование простого по структуре и легко реализуемого нечеткого регулятора с улучшенным интеллектуальным уровнем (*wise controller [11,36]*) для эффективного управления сложными управляемыми объектами (предметами).
- Согласно представленным примерам, интеллектуальное управление позволяет гарантированно и с минимальными затратами ресурсов достигать цели управления в непредсказуемых ситуациях управления, что, по сути, отражает на неформальном уровне определение целенаправленной деятельности интеллектуальных систем управления.
- Таким образом, эффективность и необходимость применения квантовых вычислений и алгоритмов управления была показана не только для квантовых систем (предложение Фейнмана), но и для классических управляемых объектов, что подтверждает результаты работ [11,27, 28, 33-36].

Список источников

1. Ульянов С. В., Николаева А. В., Решетников А. Г. Интеллектуальные системы управления в непредвиденных ситуациях. Оптимизатор баз знаний на мягких вычислениях. LAP LAMBERT Acad. Publ., OmniScriptum GmbH & Co. KG, 2013.
2. Ульянов С. В. Интеллектуальное робастное управление: технологии мягких вычислений. – М.: ВНИИГеосистем, 2011. – 408 с.
3. Ulyanov S.V, et al. System and method for nonlinear dynamic control systems based on soft computing with discrete constraints. Patent US 6,950,712 B2, 2005.
4. Интеллектуальные системы управления : учебное пособие для вузов: в 5 т. Т. 4 : Оптимизатор баз знаний на квантовых вычислениях: в 2 ч. Ч. 1. Квантовая самоорганизация баз данных и квантовый нечеткий вывод / С. В. Ульянов, Л. В. Литвинцева, В. А. Албу [и др.]. – Дубна : Международный университет природы, общества и человека "Дубна", 2014. – 224 с.
5. Интеллектуальные системы управления : учебное пособие для вузов: в 5 т. Т.3 : Оптимизатор баз знаний на мягких вычислениях: в 2 ч. Ч.1: Обучение, адаптация и моделирование / С. В. Ульянов, Л. В. Литвинцева, С. В. Сорокин [и др.]. – Дубна : Международный университет природы, общества и человека "Дубна", 2014. – 170 с.
6. Интеллектуальные системы управления: в 5 т. Т. 3. Оптимизатор баз Интеллектуальные системы управления : учебное пособие для вузов: в 5 т. Т. 3 : Оптимизатор баз знаний на

мягких вычислениях: в 2 ч. Ч. 2: Интеллектуальные регуляторы и робототехника / С. В. Ульянов, Л. В. Литвинцева, О. Ю. Тятошкина [и др.]. – Дубна : Международный университет природы, общества и человека "Дубна", 2014. – 170 с.

7. Интеллектуальные системы управления : учебное пособие для вузов: в 5 т. Т. 4 : Оптимизатор баз знаний на квантовых вычислениях: в 2 ч. Ч.2. Самоорганизующиеся интеллектуальные системы управления / С. В. Ульянов, А. В. Николаева, А. Г. Решетников, И. А. Бархатова. – Дубна : Международный университет природы, общества и человека "Дубна", 2014. – 183 с.
8. Ulyanov S.V. Self-organized quantum robust control methods and systems for situations with uncertainty and risk. Patent US 8788450 B2, 2014.
9. Ulyanov S. V., Litvintseva L. V., Ulyanov S. S. Quantum swarm model of self-organization process based on quantum fuzzy inference and robust wise control design // Proc. 7th International Conference on Application of Fuzzy Systems and Soft Computing. – 2006. – C. 13-14.
10. Ulyanov S. V. Quantum fuzzy inference based on quantum genetic algorithm: quantum simulator in intelligent robotics //International Conference on Theory and Application of Soft Computing, Computing with Words and Perceptions. – Cham : Springer International Publishing, 2019. – C. 78-85. – DOI: 10.1007/978-3-030-35249-3_92020.
11. Ulyanov, S. Intelligent Cognitive Robotics : Quantum self-organization of imperfect knowledge bases: quantum intelligent force control and information-thermodynamic law of extracted informed useful work. Vol.2 / S. Ulyanov. – Москва : Общество с ограниченной ответственностью Издательство "КУРС", 2022. – 472 с. – (Quantum IT ; Volume II).
12. Ульянов С. В., Решетников Г. П., Решетников А. Г. Технологии интеллектуальных вычислений. Квантовые вычисления и программирование в самоорганизующихся интеллектуальных системах управления. – Дубна: ОИЯИ, 2015. – 246 с.
13. Ulyanov S. V. Self-organized intelligent robust control based on quantum fuzzy inference // Recent Advances in Robust Control – Novel Approaches and Design Methods / A. Mueller (Ed.), Ch. 9. – IntechOpen, 2011. – C. 187-220.
14. Ulyanov S. V. System and method for control using quantum soft computing. Patent US 6,578,018 B1, Filed Jul. 27, 1999. Date of patent. Jun. 10,2003.
15. Litvintseva L. V., Ulyanov S. V. Quantum fuzzy inference for knowledge base design in robust intelligent controllers // Journal of Computer and Systems Sciences International.– 2007. – T. 46, № 6. – C. 908–961. – DOI:10.1134/S1064230707060081.
16. Ulyanov S .V. Self-organization of robust intelligent controller using quantum fuzzy inference // 3rd International Conference on Intelligent System and Knowledge Engineering. – IEEE, 2008. – T. 1. – C. 726–732. – DOI:10.1109/ISKE.2008.4731026.
17. Maximum work extraction and implementation costs for nonequilibrium Maxwell's demon / H. Sandberg, J. C. Delvenne, N. J. Newton, S. K. Mitter // Physical Review E. – 2014.– № 4. – C. 042119.
18. Ульянов С. В., Решетников А. Г. Синергетика информационно-когнитивного взаимодействия в интеллектуальных робототехнических системах с дистанционным обменом знаниями // Программные продукты и системы. – 2017. – № 4. – С. 593-600.
19. Ульянов С. В. Квантовая релятивистская информатика. – LAP LAMBERT Acad. Publ, OmniScriptum GmbH & Co. KG, 2015.
20. Sieniutycz S. Hamilton-Jacobi-Bellman framework for optimal control in multistage energy systems // Physics Reports. – 2000. – T. 326, № 2. – C. 165-258.
21. Ulyanov S. V. Quantum soft computing in control process design: quantum genetic algorithms and quantum neural network approaches // Proceedings World Automation Congress, 2004. – IEEE, 2004. – T. 17. – C. 99-104.
22. The improvement of quantum genetic algorithm and its application on function optimization / H. Wang, J. Liu, J. Zhi, C. Fu // Mathematical problems in engineering. – 2013. – T. 2013, №. 1. – C. 730749.

23. Malossini A., Blanzieri E., Calarco T. QGA: A quantum genetic algorithm. Technical Report # DIT-04-105, University of Toronto, 2004.
24. Lahoz-Beltra R. Quantum genetic algorithms for computer scientists // Computers. – 2016. – Т. 5, №. 4. – С. 24.
25. Ульянов С. В., Рябов Н. В. Квантовый симулятор в задачах моделирования интеллектуального нечеткого управления // Нечеткие системы и мягкие вычисления. – 2019. – Т. 14, № 1. – С. 19–33. DOI: 10.26456/fssc49.
26. Нефёдов Н. Ю., Ульянов С. В. Моделирование квантового поискового алгоритма Гровера на классическом компьютере // Системный анализ в науке и образовании. – 2010. – №2. – С. 49-64. – URL: <http://sanse.ru/download/53>.
27. Nikolaeva A. B., Ulyanov S. V. Проектирование интеллектуальной системы управления роботом манипулятором. Ч. 2: Декомпозиция управления и физический эксперимент на основе технологии мягких вычислений // Системный анализ в науке и образовании. – 2013. – №1.– С. 144-166.
28. Nikolaeva A. B., Petrov C. P., Ulyanov S. V. Проектирование интеллектуальной системы управления роботом-манипулятором. Ч. 1: Технологии мягких вычислений // Системный анализ в науке и образовании. – 2012. - № 3. – С. 73-97.
29. Nielsen M. A., Chang I. L. Quantum Computation and Quantum Information. – Cambridge university press, 2000.
30. Marinescu D. C., Marinescu G. M. Approaching quantum computing. – Pearson Prentice Hall, 2006.
31. Ульянов С. В., Сорокин С. В., Литвинцева Л. В. Оптимизатор баз знаний на основе квантовых вычислений для проектирования самоорганизующихся нечетких регуляторов: программный инструментарий // Системный анализ в науке и образовании. – 2012. – № 1. – С. 132-159.
32. Ghisi F., Ulyanov S. The information role of entanglement and interference operators in Shor's quantum algorithm gate dynamics // Journal of Modern Optics. – 2000. – Т.47, № 12. – С. 2079-2090.
33. Nikolaeva A.B., Ульянов С.В. Проектирование интеллектуальной системы управления роботом манипулятором. Ч. 3: Моделирование и физический эксперимент на основе технологии мягких вычислений // Системный анализ в науке и образовании . – 2013.– №1. – С. 167-192.
34. Ulyanov S. V. Quantum Algorithm of Imperfect KB Self-organization Pt I: Smart Control - Information-Thermodynamic Bounds // Artificial Intelligence Advances. – 2021. – Т. 3, № 2. – С. 13-36. – DOI: 10.30564/aia.v3i2.3171.
35. Ulyanov S. V., Ulyanov V. S. , Hagiwara T. Robust Quantum Controllers: Quantum Information -- Thermodynamic Hidden Force Control in Intelligent Robotics based on Quantum Soft Computing // arXiv preprint. – Дата публикации: 18.05.2023. — DOI: 10.48550/arXiv.2305.11254.
36. Intelligent cognitive robotics. Vol. III: Quantum computational toolkit of quantum self-organized intelligent control system simulator: quantum deep machine learning on quantum-inspired neural network and quantum genetic algorithm / A. G. Reshetnikov, S. V. Ulyanov, P. V. Zrelov, D. P. Zrelova. – Москва: Курс, 2023. – 432 c.