



ISSN 2071-2898 (Print)
ISSN 2071-2901 (Online)

Сорокин А.В., [Широбоков М.Г.](#)

Разработка нейронных
сетей для управления
орбитальным движением
космического аппарата с
двигателем малой тяги

Рекомендуемая форма библиографической ссылки: Сорокин А.В., Широбоков М.Г. Разработка нейронных сетей для управления орбитальным движением космического аппарата с двигателем малой тяги // Препринты ИПМ им. М.В.Келдыша. 2018. № 269. 31 с. doi:[10.20948/prepr-2018-269](https://doi.org/10.20948/prepr-2018-269)
URL: <http://library.keldysh.ru/preprint.asp?id=2018-269>

О р д е н а Л е н и н а
ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ
имени М.В.Келдыша
Р о с с и й с к о й а к а д е м и и н а у к

А.В. Сорокин, М.Г. Ширококов

Разработка нейронных сетей для управления
орбитальным движением космических аппаратов
с двигателем малой тяги

Москва — 2018

А.В. Сорокин, М.Г. Широбоков

Разработка нейронных сетей для управления орбитальным движением космических аппаратов с двигателем малой тяги

Работа посвящена созданию двух искусственных нейронных сетей (управляющей и прогнозирующей) для управления орбитальным движением космического аппарата, снабженного двигателем малой тяги. Управляющая нейронная сеть обучается решать задачу оптимального управления между двумя векторами фазового пространства. Прогнозирующая нейронная сеть обучается распространять решения в расширенном пространстве (прямых и сопряженных переменных). Разработанные нейронные сети могут быть использованы для коррекции движения аппарата вблизи номинального решения. Описываются архитектура нейронных сетей, процесс построения обучающих выборок, а также методы обучения. Приводятся результаты моделирования поведения сетей на модельных примерах.

Ключевые слова: искусственная нейронная сеть, малая тяга, орбитальное движение, принцип максимума Понтрягина

A. V. Sorokin , M. G. Shirobokov

Artificial neural networks for low-thrust spacecraft control

In this work, two artificial neural networks aimed at low-thrust spacecraft control are designed. One of the networks is called the controlling network; the another is called the predictive network. The controlling network is learned to solve the optimal control problem between two phase vectors. The predictive network is learned to propagate solutions of the extended equations of motion (prime and dual variables of the Pontryagin's maximum principle). The proposed networks can be used for orbital correction near a nominal regime. The architecture of the networks as well as sample construction process and learning methods are described. Simulation results are provided.

Key words: artificial neural network, low thrust, orbital motion, Pontryagin's maximum principle

Работа поддержана грантом Российского фонда фундаментальных исследований (РФФИ) 18-31-00403.

1. Введение

На сегодняшний день искусственные нейронные сети (ИНС) применяются в самых разных областях науки и техники. В первую очередь это задачи распознавания образов и речи, прогнозирование, теория принятия решений, построение управления. Область применения ИНС активно расширяется, они находят себя в экономике, медицине, биохимических исследованиях, всевозможных системах фильтрации информации и др.

В последнее время наблюдается отчетливый тренд на применение методов машинного обучения в механике космического полета. Вообще, идея использования искусственных нейронных сетей в задачах механики космического полета появилась сравнительно давно. Первые попытки, по-видимому, были связаны с коррекцией управления угловым движением аппарата в условиях изменения моментов инерции космических станций [1, 2] и начались в начале 1990-х годов. К настоящему моменту все работы условно можно разделить на пять направлений, посвященных 1) орбитальному движению аппарата, 2) угловому движению, 3) посадке аппарата на поверхность небесного тела, 4) диагностике систем, и 5) всем остальным темам, которые трудно классифицировать.

В направлении по орбитальному движению аппаратов заметное место занимают методы построения управления, основанные на двух биоинспирированных идеях: нейронных сетях и эволюционном программировании [3, 4, 5, 6, 7]. В этих методах проблема обучения нейронных сетей, как проблема выбора метода оптимизации, решается с использованием эволюционных алгоритмов: связи между нейронами «эволюционируют» в процессе оптимизации. Такой подход позволил добиться содержательных успехов в плане проектирования сложных многоэтапных межпланетных траекторий [7].

Среди других работ по орбитальному движению можно назвать исследования по разработке нейронных сетей для построения адаптивного управления формациями спутников [8, 9, 10]. Как было показано, нейронные сети способны парировать случайные возмущения и неточность модели даже в условиях неустойчивой динамики вокруг коллинеарных точек либрации [8] и добиваться субмиллиметровой точности поддержания формации. Дополнительной устойчивости орбитального движения можно добиться и путем синтеза нейроуправления с классическими методами стабилизации (например, со скользящим управлением [9]) и современными методами глобальной оптимизации (например, с методом роя частиц [10]).

Упомянутые выше исследования предполагают обучение нейронных сетей «на земле», т.е. на этапе до запуска космического аппарата, чтобы либо провести предварительный анализ миссии, либо поместить обученную нейронную

сеть в бортовой компьютер космического аппарата для исполнения возложенных на нее функций управления. Возможности продолжения обучения на орбите в реальных условиях движения аппарата при этом не затрагиваются. Вместе с тем именно свойство продолжать обучение в полетных условиях представляется нам естественным и интересным этапом применения нейронных сетей в механике полета. В этом направлении авторам работы известны следующие работы: по уточнению физических свойств атмосферы [11, 12, 13] и гравитационной модели малого небесного тела [14]. Известен также прототип автономного искусственного интеллекта, который может использоваться для управления угловым движением космического аппарата [15, 16].

Отметим, наконец, что значительная часть отечественных работ в области использования нейронных сетей в космической деятельности сосредоточена на средствах диагностики состояния аппаратуры на спутнике (например, [17, 18, 19, 20]). Разработка же нейронных сетей, автономно парирующих случайные возмущения и по возможности решающих нештатные ситуации, естественным образом дополнила бы имеющиеся исследования.

Настоящая работа посвящена идее создания автономной системы управления орбитальным движением космического аппарата, оснащенного двигателем малой тяги, на основе ИНС. Цель работы – разработать систему управления аппаратом с использованием двух ИНС – прогнозирующей движение аппарата в рамках некоторой модели и управляющей реактивной тягой двигателя. В работе описываются разрабатываемые ИНС, метод построения выборки для их обучения, алгоритм обучения, а также приводятся результаты расчетов. Рассмотрен также модельный пример использования разработанных сетей к задаче перелета между геопереходной и геостационарной орбиты.

2. Постановка задачи

Для того, чтобы поставить задачу, введем вспомогательные предположения, переменные и уравнения. В данной работе рассматривается орбитальное движение космического аппарата (КА). Аппарат снабжен двигателем малой тяги и движется в центральном гравитационном поле Земли согласно уравнениям движения:

$$\begin{cases} \dot{\mathbf{r}} = \mathbf{v}, \\ \dot{\mathbf{v}} = -\frac{\mu\mathbf{r}}{r^3} + \mathbf{u}, \end{cases}$$

где \mathbf{r} – положение аппарата относительно Земли, \mathbf{v} – скорость аппарата, $r = |\mathbf{r}|$, μ – гравитационный параметр Земли, \mathbf{u} – реактивное ускорение, сообщаемое

двигателем. Ограничения на величину и направление реактивного ускорения не накладываются.

Пусть требуется перевести аппарат из точки \mathbf{r}_1 , где он имеет скорость \mathbf{v}_1 , в точку \mathbf{r}_2 , где он должен иметь скорость \mathbf{v}_2 , за время T . Среди всевозможных функций управления \mathbf{u} нас будет интересовать та, которая доставляет минимум функционалу

$$J = \frac{1}{2} \int_0^T u^2 dt \rightarrow \min.$$

Вообще надо отметить, что решение возникающей задачи оптимального управления часто сводят к решению системы нелинейных уравнений (двухточечной краевой задачи), возникающей в результате использования принципа максимума Понтрягина. Так поступим и мы: введем в рассмотрение сопряженные переменные $\boldsymbol{\lambda}_r$ и $\boldsymbol{\lambda}_v$ и гамильтониан

$$H = -\frac{1}{2}u^2 + \boldsymbol{\lambda}_r^T \mathbf{v} + \boldsymbol{\lambda}_v^T \left(-\frac{\mu \mathbf{r}}{r^3} + \mathbf{u} \right).$$

Максимизируя гамильтониан относительно переменной \mathbf{u} , получаем функцию управления $\mathbf{u}^* = \boldsymbol{\lambda}_v$. Теперь запишем расширенные уравнения движения

$$\begin{cases} \dot{\mathbf{r}} = \mathbf{v}, \\ \dot{\mathbf{v}} = -\frac{\mu \mathbf{r}}{r^3} + \boldsymbol{\lambda}_v, \\ \dot{\boldsymbol{\lambda}}_r = \frac{\mu \boldsymbol{\lambda}_v}{r^3} - \frac{3\mu \mathbf{r}^T \boldsymbol{\lambda}_v}{r^5} \mathbf{r}, \\ \dot{\boldsymbol{\lambda}}_v = -\boldsymbol{\lambda}_r. \end{cases} \quad (1)$$

Как обычно, любое управление $\mathbf{u}^*(t)$, удовлетворяющее необходимым условиям принципа максимума Понтрягина, т.е. начальным условиям $\mathbf{r}(0) = \mathbf{r}_1$, $\mathbf{v}(0) = \mathbf{v}_1$ и краевым условиям $\mathbf{r}(T) = \mathbf{r}_2$, $\mathbf{v}(T) = \mathbf{v}_2$, будем считать оптимальным. Это управление определяется неизвестными начальными условиями $\boldsymbol{\lambda}_r(0)$, $\boldsymbol{\lambda}_v(0)$, которые при интегрировании уравнений (1) вместе с условиями $\mathbf{r}(0) = \mathbf{r}_1$, $\mathbf{v}(0) = \mathbf{v}_1$ приводят к равенствам $\mathbf{r}(T) = \mathbf{r}_2$, $\mathbf{v}(T) = \mathbf{v}_2$. Отметим, что функция управления $\mathbf{u}^*(t) = \boldsymbol{\lambda}_v(t)$ определяется из решения уравнений (1) конечным набором чисел – сопряженными переменными в начальный момент времени. Найти начальные условия $\boldsymbol{\lambda}_r(0)$, $\boldsymbol{\lambda}_v(0)$ можно с помощью метода продолжения по параметру (см., например, [21]). Число витков N_{rev} вокруг Земли при этом может быть выбрано произвольно.

Итак, пусть для конкретных $\mathbf{r}_1, \mathbf{v}_1, \mathbf{r}_2, \mathbf{v}_2, T$ и N_{rev} тем или иным образом было найдено оптимальное управление $\mathbf{u}^*(t)$, которое мы назовем номинальным управлением, а траекторию, которую оно определяет, номинальной траекторией.

Данная работа решает две задачи. Первая задача состоит в построении ИНС, аппроксимирующей отображение $(\mathbf{r}_1, \mathbf{v}_1, \mathbf{r}_2, \mathbf{v}_2, T) \rightarrow (\boldsymbol{\lambda}_r(0), \boldsymbol{\lambda}_v(0))$ вблизи номинальной траектории. Эту нейронную сеть мы называем *управляющей* нейронной сетью. Вторая задача состоит в построении ИНС, аппроксимирующей отображение $(\mathbf{r}(0), \mathbf{v}(0), \boldsymbol{\lambda}_r(0), \boldsymbol{\lambda}_v(0), T) \rightarrow (\mathbf{r}(T), \mathbf{v}(T), \boldsymbol{\lambda}_r(T), \boldsymbol{\lambda}_v(T))$ вблизи номинальной траектории. Эту нейронную сеть мы называем *прогнозирующей* нейронной сетью.

В последующих главах работы приводится описание архитектуры разрабатываемых нейронных сетей, метод формирования выборки для их обучения, метод обучения, а также некоторые результаты, касательно применения данных нейронных сетей к задаче управления аппаратом при многовитковом перелете между геопереходной и геостационарной орбитами.

3. Введение в нейронные сети

С математической точки зрения, нейронная сеть — это взвешенный ориентированный граф, в узлах которого находятся функции, называемые активационными. Процесс изменения весов в графе называется ее обучением. Формально обучение нейронных сетей — это многопараметрическая задача нелинейной оптимизации. Существует несколько способов обучения ИНС, таких как обучение с учителем, обучение без учителя и обучение с подкреплением. Каждый способ применяется в своем классе задач. Обучение с учителем позволяет решать задачу регрессии и классификации объектов. В этом случае для настройки параметров сеть использует массив данных, состоящий из входных сигналов и соответствующих им выходных сигналов. Метод обучения нейронных сетей без учителя позволяет решать задачу кластеризации, поиска ассоциаций и сокращения размерности. Алгоритмы обучения с подкреплением выдают стратегию, приписывающую состояниям окружающей среды действия, которые должна предпринять система в этих состояниях. В данном разделе коротко рассматриваются методы обучения с учителем, соответствующие нейронные сети и их архитектуры.

Структура графа ИНС определяется следующим образом: выделяется множество вершин, называемых входным слоем нейронной сети. Далее в граф определенным образом добавляется еще одно множество вершин, называемое скрытым слоем, после чего определяются ребра, связывающие два слоя. Процедура

добавления скрытых слоев итеративна, то есть таким образом можно построить N скрытых слоев. Последний слой нейронной сети называется выходным. После построения графа произвольным образом инициализируются веса в его ребрах. На рисунке 1 приведен пример многослойной нейронной сети.

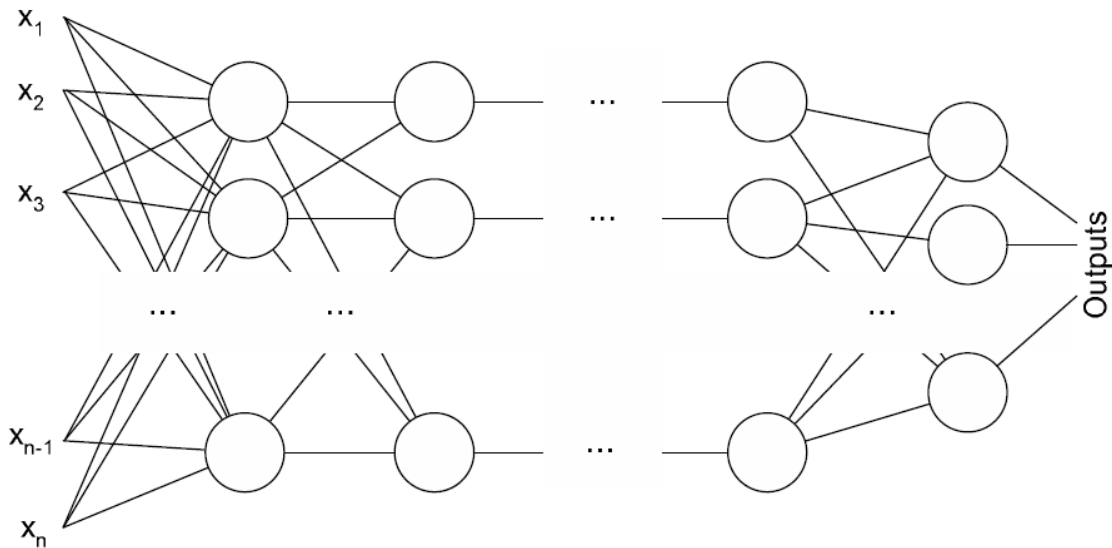


Рис. 1. Многослойная нейронная сеть.

В каждом узле графа размещена некоторая функция единственного аргумента. Данная функция называется активационной и обладает свойствами монотонности и дифференцируемости. Пример активационной функции — сигмоида:

$$\phi(x) = \frac{1}{1 + e^{-x}}, \quad \phi'(x) = \phi(x) \cdot (1 - \phi(x)).$$

Каждый слой сети независимо друг от друга обладает одной и той же активационной функцией. После того как была определена структура нейронной сети, через входной слой по сети распространяется сигнал. Под входным сигналом будем понимать некоторый вектор, длина которого совпадает с количеством нейронов входного слоя сети. Выходной сигнал — это вектор, длина которого равна количеству нейронов в выходном слое сети. В каждом узле графа вычисляется значение активационной функции от линейной комбинации сигнала. Поскольку веса в графе в начальный момент были определены произвольно, то значения в выходном слое также будут произвольными. Обучение с учителем предполагает наличие обучающей выборки типа входной-выходной сигнал. Под обучающей выборкой будем понимать конечный набор из входных и соответствующих им выходных сигналов. После того как входной сигнал распространился по сети, необходимо ввести некоторый функционал ошибки, для того

чтобы оценить меру совпадения результата работы сети — предсказанного сигнала, и соответствующего выходного сигнала. Существует множество подобных функционалов, к примеру, средний модуль отклонения:

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \tilde{y}_i|,$$

где y_i — i -я компонента выходящего сигнала, \tilde{y}_i — i -я компонента предсказанного сигнала, а N — количество компонент выходного сигнала. Поскольку сигнал — это вектор, то под его компонентами будем понимать векторные компоненты. В итоге принцип работы ИНС заключается в том, чтобы минимизировать имеющийся функционал посредством изменения весов в графе. Большинство методов оптимизации являются градиентными, именно поэтому накладывалось условие дифференцируемости на активационные функции. Одним из основных методов минимизации функционалов ошибки в ИНС, который используется в данной работе, является метод обратного распространения ошибки. Рассмотрим принцип работы этого алгоритма на примере простейшего функционала ошибки, приведенного ранее — $MAE = \frac{1}{N} \sum_{i=1}^N |d_i - t_i|$, где d_i — i -я компонента выходящего сигнала, t_i — i -я компонента предсказанного сигнала, а N — количество компонент выходного сигнала. Введем необходимые обозначения. Пусть пример обучения представлен парой (\mathbf{x}, \mathbf{d}) , где \mathbf{x} — входной вектор, а \mathbf{d} — желаемый отклик. Пусть в сети L слоев, тогда для j -го нейрона l -го слоя линейная комбинация выглядит следующим образом: $v_j^l = \sum_{i=0}^M w_{ji}^l \cdot y_i^{l-1}$, где y_i^{l-1} — выходной сигнал нейрона i , расположенного в слое $l-1$, w_{ji}^l — вес связи нейрона j слоя l с нейроном i слоя $l-1$. В случае сигмоидальной функции активации выходной сигнал нейрона j слоя l имеет следующий вид: $y_j^l = \phi_j(v_j^l)$. Если $l=0$, т.е. нейрон j находится во входном слое, то $y_j^0 = x_j$, где x_j — j -я компонента входного вектора \mathbf{x} . Если же нейрон j находится в выходном слое, то $y_j^L = t_j$. Таким образом, j -я компонента функционала ошибки равна $e_j = |d_j - t_j|/N$. Для минимизации ошибки будем двигаться в направлении антиградиента. Значения локальных градиентов δ_j^l в узлах графа задаются следующей формулой:

$$\delta_j^l = \begin{cases} e_j^L \cdot \phi_j'(v_j^L) & \text{для нейрона } j \text{ выходного слоя } L \\ \phi_j'(v_j^l) \cdot \sum_k \delta_k^{l+1} \cdot \delta_{kj}^{l+1} & \text{для нейрона } j \text{ скрытого слоя } l \end{cases}$$

Следовательно, изменение весов в l -м слое будет происходить по следующей формуле: $w_{ji}^l \leftarrow w_{ji}^l + \lambda \cdot \delta_j^l \cdot y_i^{l-1}$. Процесс итеративный и завершается в том случае, когда функционал ошибки достигает определенного значения.

Качество обучения нейронных сетей напрямую зависит от выбора функционала ошибки, метода оптимизации и архитектуры сети.

4. Решение задачи перелета с малой тягой

В данной работе авторами был реализован в среде MATLAB метод продолжения по параметру — функция `EnergyOptimalProblem`, принимающая в качестве аргументов значения начального и конечного фазовых векторов \mathbf{x}_1 и \mathbf{x}_2 , время, выделенное для перелета t , и количество витков N_{rev} относительно притягивающего центра. Результатом ее работы является вектор сопряженных переменных в начальный момент времени. В постановке задачи рассматривался оптимальный перелет из точки в точку в рамках невозмущенной задачи двух тел, теперь же добавим в систему уравнений движения возмущение \mathbf{F} . Рассмотрим решение возмущенной задачи с помощью функции `EnergyOptimalProblem`. Идея решения задачи оптимального перелета в рамках возмущенной задачи двух тел заключается в разбиении возмущенной траектории на отрезки, на каждом из которых будет рассматриваться оптимальный перелет в рамках невозмущенной задачи двух тел. Для этого введем некоторый промежуток времени Δt , не зависящий от длительности перелета. При интегрировании возмущенной траектории спустя каждый промежуток времени Δt будем проверять евклидову норму разности радиус-вектора конечной точки и радиус-вектора конечной точки, полученного путем интегрирования. Интегрирование производится методом Рунге—Кутты 4-го порядка с переменным шагом. В случае, когда значение нормы будет превышать заранее заданную величину ε , будем рассчитывать значение вектора сопряженных переменных, необходимое для оптимального перелета в конечную точку. Для этого определим значения аргументов функции расчета сопряженных переменных: первый аргумент будет меняться в зависимости от времени, его значение — текущий фазовый вектор, второй аргумент — конечный фазовый вектор, который не меняется на протяжении всего полета, третий аргумент — время полета из текущей точки в конечную. Количество витков также не меняется.

Теперь перейдем к построению решения с помощью нейронных сетей. Осуществим переход следующим образом: вместо интегрирования невозмущенной траектории методом Рунге—Кутты построим прогнозирующую нейронную сеть, которая по значению текущего фазового вектора будет рассчитывать значение конечного фазового вектора. Также построим управляющую нейронную сеть, которая будет рассчитывать значение вектора сопряженных переменных в случае, когда евклидова норма разности фазового вектора конечной точки и фазового вектора конечной точки, полученного прогнозирующей сетью, будет пре-

вышать величину ε .

Перейдем к составлению обучающих выборок. Если для задачи прогнозирования численное решение можно построить для любой точки фазового пространства, то в случае нейронной сети необходимо определить ограниченную область фазового пространства, в которой будет составлена выборка. В качестве этой области будем брать некоторую трубку траекторий. Определим два шара, первый — радиуса ε_1 с центром в точке, задаваемой начальным радиус-вектором, второй — радиуса ε_2 с центром в точке, задаваемой вектором начальной скорости. Составим в этих шарах конечный набор начальных условий следующим образом: компоненты радиус-вектора будем выбирать из равномерного распределения с нулевым математическим ожиданием на шаре радиуса ε_1 . Компоненты вектора скорости выберем из того же распределения, но на шаре радиуса ε_2 . Также необходимо определить значения векторов сопряженных переменных для каждого из варьируемых фазовых векторов. Они будут рассчитываться с помощью функции `EnergyOptimalProblem`. Теперь определим вектор состоящий из фазового и сопряженного векторов. Положив его в качестве начального условия для системы уравнений (1) можно однозначно построить траекторию, удовлетворяющую данной системе. Для каждого такого начального условия проинтегрируем с переменным шагом соответствующую траекторию в рамках невозмущенной модели. Таким образом, для нейронной сети, прогнозирующей движение, обучающая выборка будет состоять из фазовых и сопряженных векторов, отвечающих траекториям, текущего времени полета, и соответствующих им конечных фазовых векторов: $S = \{S_i\}_{i=1}^M$, где $S_i = \{[\mathbf{X}_n^i, \boldsymbol{\lambda}_n^i, t_n, \mathbf{X}_{end}^i]\}_{n=1}^N$, M — количество траекторий, а N — количество точек на траектории. На рисунке 2 приведен набор траекторий в координатном пространстве для обучающей выборки прогнозирования.

Для формирования обучающей выборки для задачи коррекции, необходимо аппроксимировать нейронной сетью отображение, задаваемое функцией `EnergyOptimalProblem`. Данное отображение также определим в ограниченной области фазового пространства. Для этого возьмём номинальную траекторию и построим относительно нее трубку фиксированного радиуса δ в координатном пространстве. Равномерно распределим в ней конечное количество радиус-векторов КА. Также необходимо варьировать значения векторов скоростей. Для этого будем добавлять к исходной скорости вектор произвольного направления значение нормы которого определим как один процент от нормы исходной скорости. Для каждого такого фазового вектора посчитаем значения сопряженных переменных, необходимые для того, чтобы совершить оптимальный по энергии перелет в конечную точку. Таким образом, для нейронной сети, корректирующей движение, обучающая выборка будет состоять из предложенных фазо-

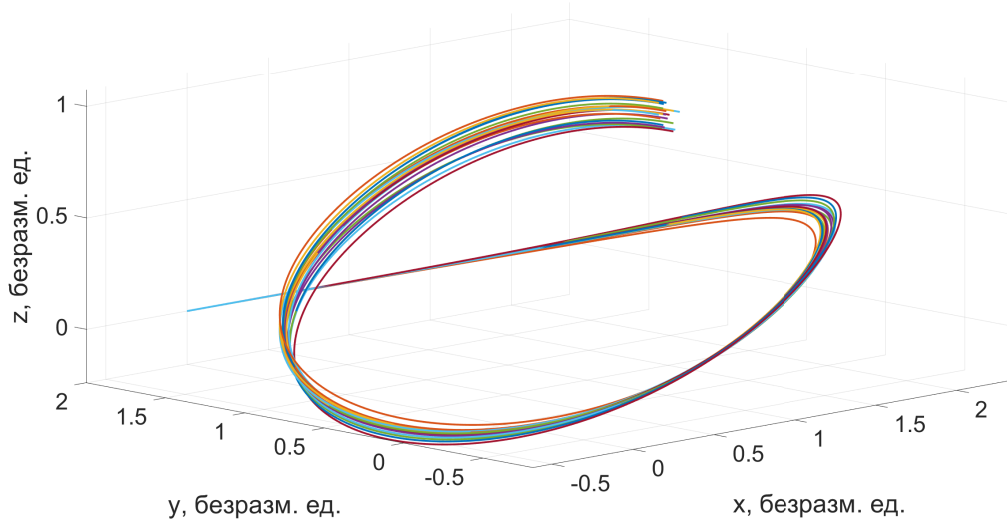


Рис. 2. Координатное представление траекторий для выборки прогнозирования.

вых векторов, времени полета и соответствующих им сопряженных переменных: $S = \{[\mathbf{X}_i, t_i, \boldsymbol{\lambda}_i]\}_{i=1}^N$, где N — количество заданных позиций. На рисунке 3 приведена визуализация обучающей выборки для задачи коррекции в координатном пространстве.

После того как обучающие выборки составлены, необходимо построить нейронные сети, способные их аппроксимировать. Обе нейронные сети должны решать задачу многомерной нелинейной регрессии, другими словами, для каждой выборки необходимо построить аппроксимирующее отображение. Таким образом, выбор архитектуры сети и функционалов ошибки значительно сужается. В 1957 году А.Н. Колмогоровым и В.И. Арнольдом была доказана теорема о представлении непрерывных функций нескольких переменных суперпозицией непрерывных функций одной переменной, которая в 1989 году была переформулирована и доказана Д. Цыбенко для нейронных сетей: любая функция нескольких переменных может быть представлена трехслойной ИНС с прямыми полными связями между входным и скрытым слоями с N нейронами входного слоя, не менее чем $2N + 1$ нейронами скрытого слоя с ограниченными функциями активации и нейронами выходного слоя с неизвестными функциями активации. Следовательно, архитектура сети будет задана следующим образом: полный граф с 3 слоями: входным, скрытым и выходным. Функционал ошибки выберем из физических соображений. Поскольку обе нейронные сети должны минимизировать евклидово расстояние, то в качестве функционала выберем среднеквадратичное

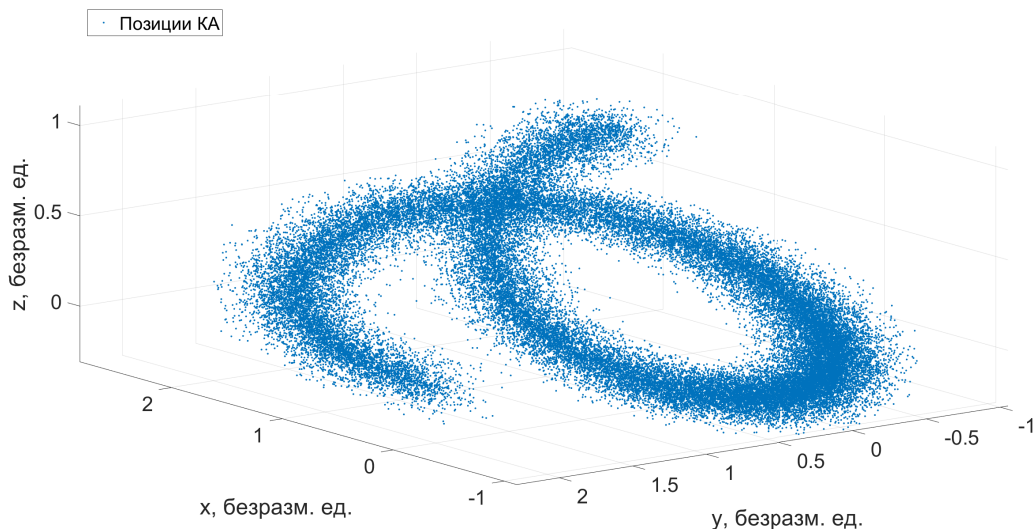


Рис. 3. Координатное представление обучающей выборки для коррекции.

отклонение:

$$MSE = \frac{1}{N} \sum_{n=1}^N (y_n - \tilde{y}_n)^2,$$

где N — количество примеров в обучающей выборке, y_n — выходной обучающий пример, а \tilde{y}_n — желаемый отклик сети. Теперь определимся с активационными функциями. Для скрытого слоя в качестве нелинейной активационной функции выберем плотность нормального распределения. Она является дифференцируемой и монотонной, поэтому удовлетворяет определению активационной функции. Полученная нейронная сеть называется радиально-базисной нейронной сетью. Функционировать данная сеть будет в соответствии с данной формулой

$$f(\mathbf{x}|\boldsymbol{\theta}) = \omega_0 + \sum_{n=1}^M \omega_n \exp\left(-\frac{\|\mathbf{x} - \boldsymbol{\mu}_n\|^2}{2\sigma_n^2}\right),$$

где вектор параметров $\boldsymbol{\theta} = [\boldsymbol{\omega}, \boldsymbol{\mu}, \boldsymbol{\sigma}]$, а $\boldsymbol{\mu} = [\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_M]$. Здесь M — это количество нейронов в скрытом слое. Инициализация параметров будет происходить следующим образом: коэффициенты линейной комбинации ω_n задаются из равномерного распределения с нулевым математическим ожиданием, а дисперсия полагается единичной. Вектора математического ожидания определим из физического смысла следующим образом: разобьем аппроксимируемое множество — обучающую выборку — на кластеры. Под кластером подразуме-

вается плотное скопление объектов выборки. Количество кластеров определяется количеством нейронов в скрытом слое сети. Для каждого кластера определим его центр масс с помощью метода k -средних. Действие алгоритма таково, что он стремится минимизировать суммарное квадратичное отклонение точек кластеров от центров этих кластеров. Таким образом, вектора математического ожидания будут выбраны как центры масс своих кластеров. Физический смысл заключается в том, что каждый нейрон будет аппроксимировать свой кластер плотностью нормального распределения с заданными параметрами. В итоге гиперповерхность, задаваемая выборкой, будет аппроксимирована суперпозицией плотностей нормального распределения. Подобная инициализация параметров является хорошим начальным приближением для градиентного метода, оптимизирующего эти параметры. В качестве градиентного метода будет выбран алгоритм Левенберга–Марквардта. Подробно рассмотрим процесс оптимизации функции потерь. Распишем более подробно вектор параметров сети: $\boldsymbol{\theta} = [\boldsymbol{\omega}, \boldsymbol{\mu}, \boldsymbol{\sigma}]$. Его длина равна $Md + 2M + 1$, где d – размерность выборки, поскольку $\boldsymbol{\mu} = [\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_M]$ – это вектор составленный из векторов математических ожиданий всех кластеров, $\boldsymbol{\omega} = [\omega_0, \omega_1, \dots, \omega_M]$, а $\boldsymbol{\sigma} = [\sigma_1, \sigma_2, \dots, \sigma_M]$. Запишем линейное приближение аппроксимирующей функции: $f(\mathbf{x}|\boldsymbol{\theta} + \Delta\boldsymbol{\theta}) - f(\mathbf{x}|\boldsymbol{\theta}) = J\Delta\boldsymbol{\theta}$, где J – якобиан функции $f(\mathbf{x}|\boldsymbol{\theta})$ в точке $\boldsymbol{\theta}$ размерности $(N \times Md + 2M + 1)$, N – объем обучающей выборки,

$$J = \begin{bmatrix} \frac{\partial f(\mathbf{x}_1, \boldsymbol{\theta})}{\partial \theta_1} & \dots & \frac{\partial f(\mathbf{x}_1, \boldsymbol{\theta})}{\partial \theta_{Md+2M+1}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f(\mathbf{x}_N, \boldsymbol{\theta})}{\partial \theta_1} & \dots & \frac{\partial f(\mathbf{x}_N, \boldsymbol{\theta})}{\partial \theta_{Md+2M+1}} \end{bmatrix}$$

Для нахождения приращения $\Delta\boldsymbol{\theta}$ приравняем нулю вектор частных производных функционала ошибки по параметрам, для этого представим его следующим образом: $MSE(\boldsymbol{\theta}) = \|\mathbf{y} - f(\boldsymbol{\theta} + \Delta\boldsymbol{\theta})\|^2$, где $\mathbf{y} = [y_1, \dots, y_N]^T$, а $f(\boldsymbol{\theta} + \Delta\boldsymbol{\theta}) = [f(x_1|\boldsymbol{\theta} + \Delta\boldsymbol{\theta}), \dots, f(x_N|\boldsymbol{\theta} + \Delta\boldsymbol{\theta})]^T$. Преобразуем функционал: $\|\mathbf{y} - f(\boldsymbol{\theta} + \Delta\boldsymbol{\theta})\|^2 = (\mathbf{y} - f(\boldsymbol{\theta} + \Delta\boldsymbol{\theta}))^T (\mathbf{y} - f(\boldsymbol{\theta} + \Delta\boldsymbol{\theta})) = \mathbf{y}^T \mathbf{y} - 2\mathbf{y}^T f(\boldsymbol{\theta} + \Delta\boldsymbol{\theta}) + f^T(\boldsymbol{\theta} + \Delta\boldsymbol{\theta}) f(\boldsymbol{\theta} + \Delta\boldsymbol{\theta})$. Дифференцируя полученное выражение по параметрам, получим

$$\frac{\partial MSE(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = (J^T J)\Delta\boldsymbol{\theta} - J^T(\mathbf{y} - f(\boldsymbol{\theta})) = 0.$$

Таким образом, $\Delta\boldsymbol{\theta} = (J^T J)^{-1} J^T(\mathbf{y} - f(\boldsymbol{\theta}))$. Матрица $J^T J$ может оказаться вырожденной, поэтому вводят параметр регуляризации $\lambda \geq 0$

$$\Delta\boldsymbol{\theta} = (J^T J + \lambda E)^{-1} J^T(\mathbf{y} - f(\boldsymbol{\theta})),$$

где E – единичная матрица подходящей размерности. Полученное приращение параметра θ есть приращение для всех весов в нейронной сети.

5. Моделирование

Перед тем как приступить к моделированию решения, необходимо более подробно рассмотреть построение обучающей выборки для задачи коррекции. В связи с тем, что расчет значений сопряженных переменных с помощью функции `EnergyOptimalProblem` связан с большими временными затратами, для формирования выборки для задачи коррекции в работе использовалась функция дифференциальной коррекции `LevenbergEnergyOptimalProblem`. Данная функция принимает в качестве аргументов те же параметры, что и функция `EnergyOptimalProblem`, а также начальное приближение для вектора сопряженных переменных. Так как значения фазовых векторов выборки по построению лежат в окрестности номинальной траектории, наличие начального приближения значительно ускоряет процесс отыскания соответствующих векторов сопряженных переменных. Теперь рассмотрим проблемы появившиеся при формировании выборки. После составления выборки необходимо было проверить её на предмет ошибок и выбросов. Поскольку метод отыскания сопряженных переменных может, вообще говоря и не сходиться, то необходимо было исключить из выборки такие значения, которые не обеспечивали бы оптимальный перелет в конечную точку. Данная процедура выполнялась следующим образом: в случае несовпадения конечной точки номинальной траектории и конечной точки траектории, полученной интегрированием уравнений движения с начальным условием взятым из выборки, значение этого элемента исключалось из выборки. Под выбросом будем понимать элемент выборки у которого норма вектора сопряженных переменных будет находиться на расстоянии большем чем три стандартных отклонения от значения нормы, соответствующей вектору сопряженных переменных на номинальной траектории. Выбросы также исключались из выборки. Далее, по мере приближения КА к конечной точке - значения сопряженных переменных начинали экспоненциально расти по норме. Данный факт наблюдался для траекторий любого вида и продолжительности. Экспоненциальный рост нормы фиксировался для полета, время которого было менее 5 минут. Было принято решение отделить от общей выборки соответствующую часть. В случае когда количество витков оказывалось ненулевым, часть траектории пролегла рядом с притягивающим центром в следствии чего скорость КА увеличивалась на этих участках, а радиус-вектор уменьшался. При этом оказалось, что норма вектора λ_r пропорциональна норме вектора скорости, а норма вектора λ_v пропорциональна норме радиус-вектора. На рисунках 4 и 5

представлены изменения нормы векторов λ_r и λ_v , соответственно.

Выяснилось, что с ростом значений скорости возрастает и дисперсия для сопряженных переменных. На рисунке 6 представлено поведение дисперсии для нормы вектора сопряженных переменных.

При увеличении дисперсии, очевидно, ухудшается точность аппроксимации выборки нейронной сетью, поэтому было принято решение разделить выборку еще на 2 части и для каждой провести свою нормализацию. Разбиение производилось по дисперсии, то есть выборка делилась на часть с дисперсией меньшей заданной величины σ и часть с большей. Далее производилась нормализация всех компонент выборки в соответствии со следующей формулой: $y'_i = (y_i - M(\mathbf{y})) / \sqrt{D(\mathbf{y})}$, где \mathbf{y} – вектор выборки, $M(\mathbf{y})$ – выборочное математическое ожидание, посчитанное по выборке \mathbf{y} , а $D(\mathbf{y})$ – выборочная дисперсия, посчитанная по выборке \mathbf{y} . Для выборки с дисперсией большей σ производилось еще одно преобразование, а именно ее отображение с помощью арктангенса. После нормализации выборки математическое ожидание каждого столбца равняется нулю, таким образом арктангенс будет являться сжимающим отображением. Данное отображение является непрерывным и дифференцируемым, поэтому обратный переход к исходной выборке будет однозначным.

Будем называть совокупность двух обученных нейронных сетей моделью. После составления выборок и обучения нейронных сетей остается валидировать полученные результаты. Для этого проверим корректность работы модели для различных траекторий при различных возмущениях, оценить количество коррекций сопряженных переменных в зависимости от силы возмущения, а также проверить устойчивость полученного решения к начальному приближению.

Для каждой траектории, задаваемой своими граничными условиями, необходимо обучать свою модель. Начнем с тестирования работы набора моделей для различных траекторий и возмущений. Для этого произвольным образом определим пять краевых задач и для каждой обучим модель. Количество нейронов и минимальный порог функционала ошибки определим для всех моделей одинаковыми. Положим $M = 64$ нейронов, а $MSE_{threshold} = 10^{-5}$. В качестве возмущений будем брать возмущения, не меняющиеся во времени, и переменные возмущения одного и того же порядка малости. Для оценки качества работы модели введем следующую метрику – норма разности конечных радиус-векторов, отвечающих номинальной траектории и траектории, полученной нейронной сетью. Для краткости назовем ее Q -метрикой. Точность прогнозирования положим равной 100 метрам. На рисунке 7 приводится зависимость Q -метрики от порядка возмущения для каждой модели. Каждая модель тестировалась на возмущениях постоянного направления. Порядок возмущений лежал в диапазоне $10^{-8} - 10^{-5}$ безразмерных единиц.

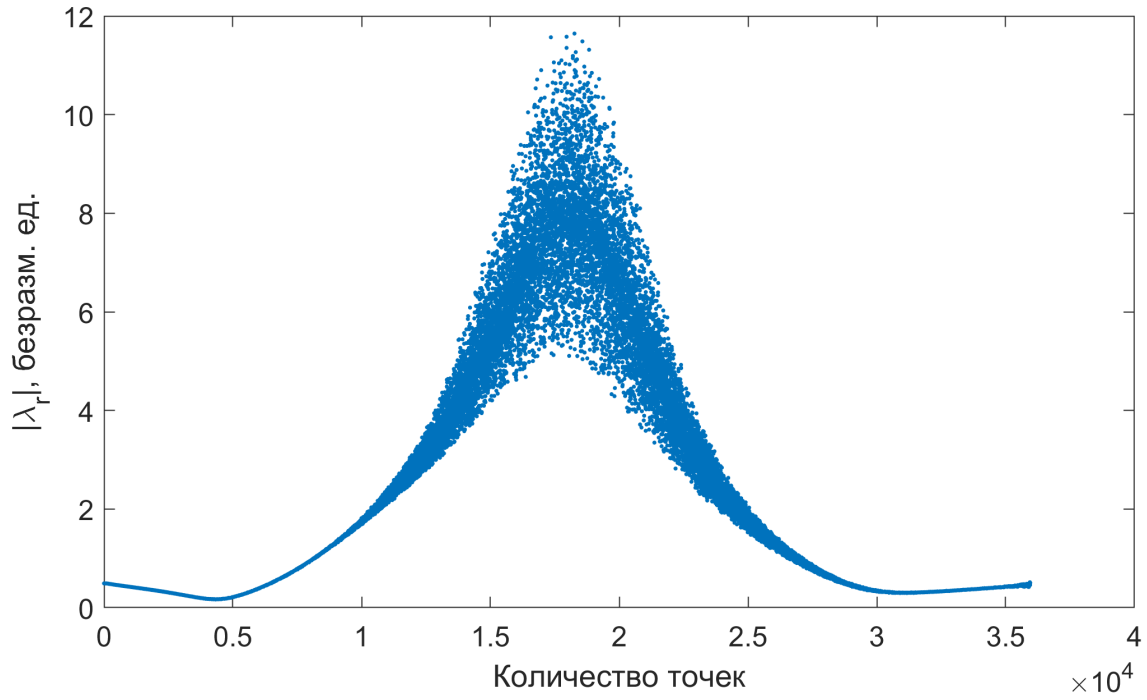


Рис. 4. Норма вектора, сопряженного к радиус-вектору.

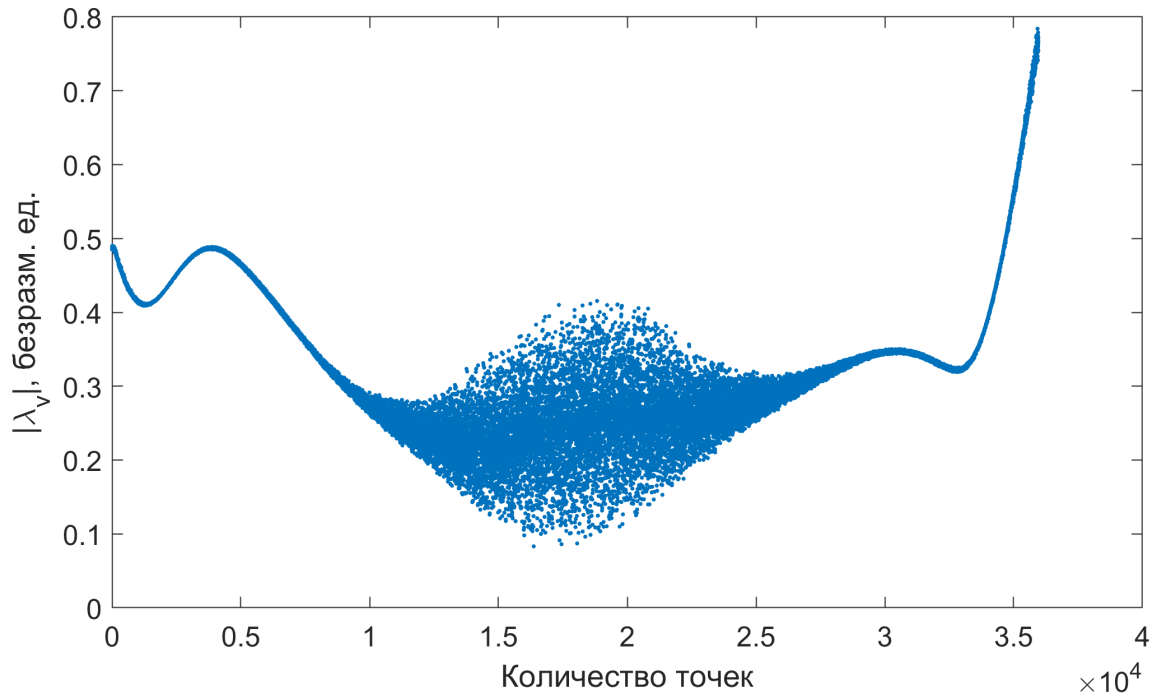


Рис. 5. Норма вектора, сопряженного к вектору скорости.

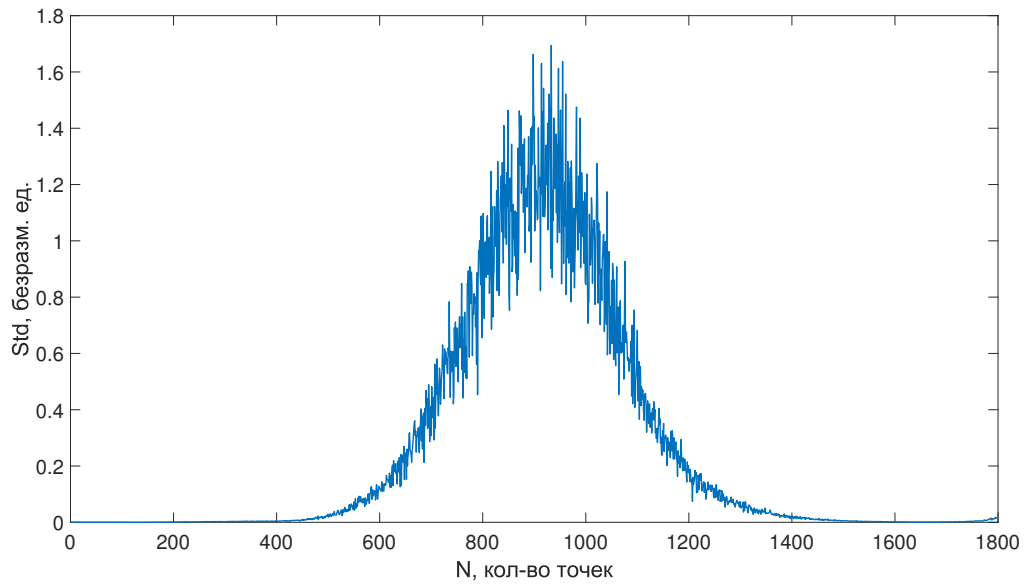


Рис. 6. Дисперсия нормы вектора сопряженных переменных.

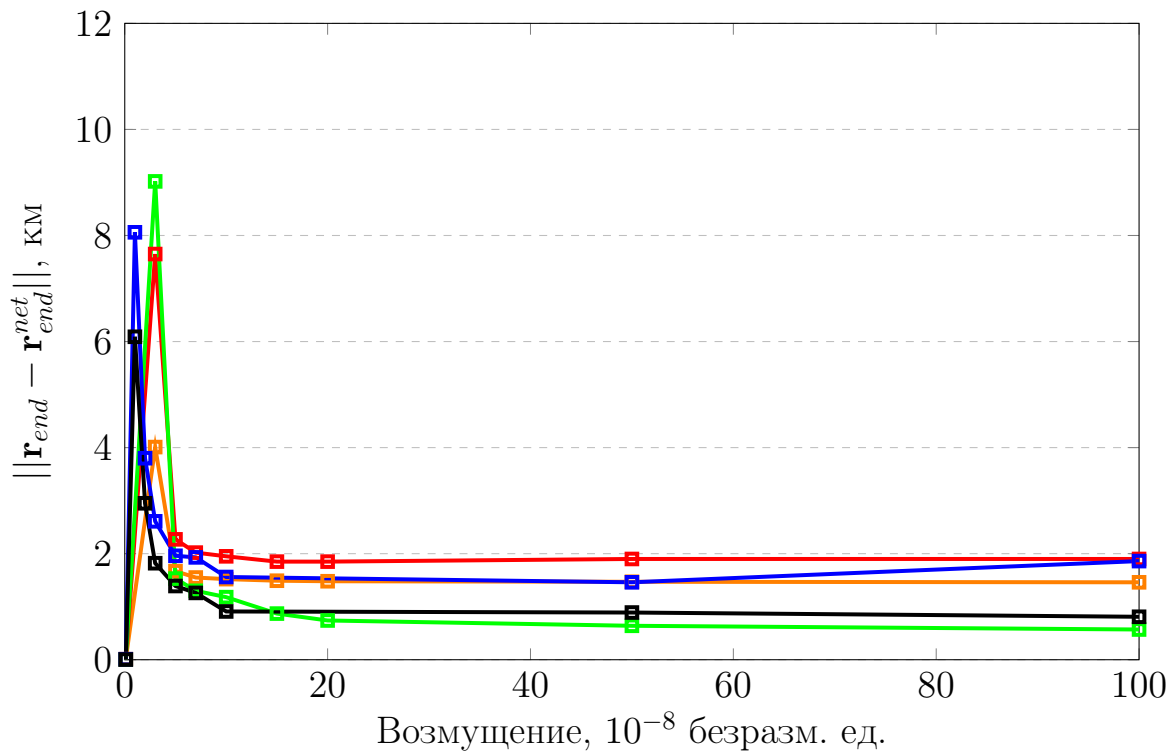


Рис. 7. Невязка по положению в зависимости от порядка возмущения.

Данный пик можно объяснить тем, что при возмущениях порядка 10^{-7} безразмерных единиц количество коррекций сопряженных переменных не превышает нескольких раз, следовательно, можно сделать вывод, что нескольких коррекций не хватает для того, чтобы добиться такой же точности, как и в случае многих коррекций. Возникает вопрос: что нужно сделать, чтобы нескольких коррекций хватало для того, чтобы обеспечить такую же точность попадания в цель, как и в случае многих коррекций? Можно сделать предположение, что при увеличении числа нейронов в нейронной сети точность попадания для нескольких коррекций увеличится. Проверим данное предположение после анализа моделей на различных возмущениях.

Теперь построим рисунок 8, на котором отложим зависимости количества коррекций сопряженных переменных от порядка возмущений.

Количество коррекций сопряженных переменных растет при увеличении силы возмущения. Теперь проверим работу тех же моделей, но уже для переменных возмущений. Для этого изобразим на рисунке 9 результат работы моделей в зависимости от порядка возмущений.

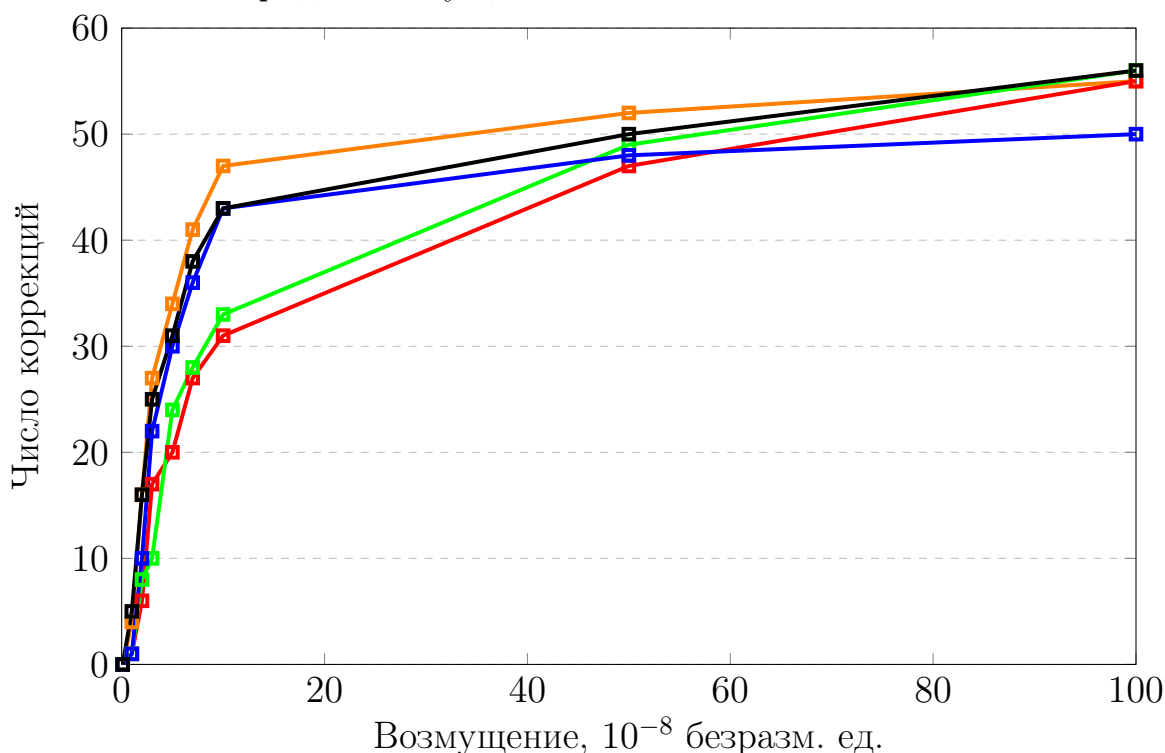


Рис. 8. Число коррекций в зависимости от силы возмущения.

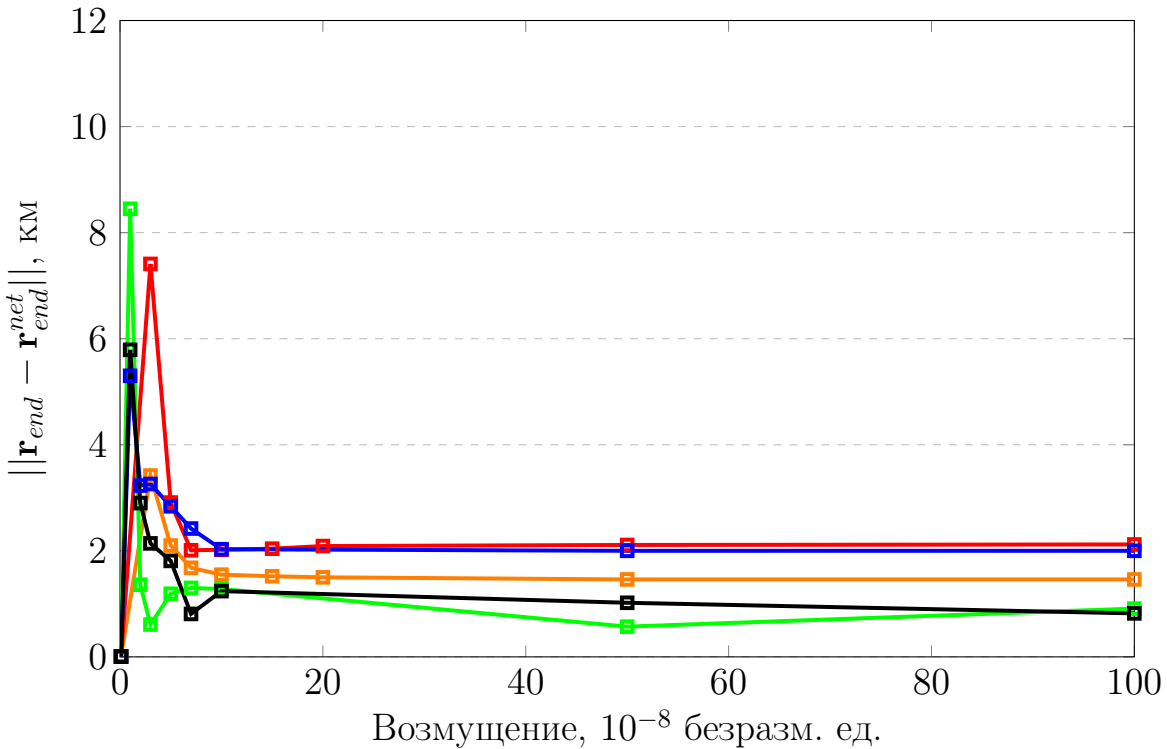


Рис. 9. Невязка по положению в зависимости от порядка возмущения.

Сравнение результатов работы моделей при переменных и постоянных возмущениях показывает незначительное расхождение в терминах определенной метрики качества, а это означает, что модели способны с некоторой точностью парировать любой из рассмотренных типов возмущения. Теперь построим рисунок 10, на котором изобразим зависимость количества коррекций от порядка в случае переменных возмущений.

Действительно, кривые коррекций для всех моделей совпадают. Это объясняется тем, что случайные возмущения не имеют постоянного направления, поэтому возмущенная траектория будет колебаться относительно невозмущенной, таким образом, количество раз, когда прогнозирующая сеть предскажет выход конечной точки траектории из допустимого шара, будет примерно одинаковым. Теперь проверим влияние количества нейронов на точность коррекции сопряженных переменных. Для этого построим пять моделей для одной и той же обучающей выборки. Число нейронов для моделей определим следующим образом: $M = 16, 32, 64, 128, 256$. Если скачок Q -метрики будет сглаживаться при увеличении количества нейронов, то предположение окажется верным. Изобразим на рисунке 11 зависимость Q -метрики от силы возмущения.

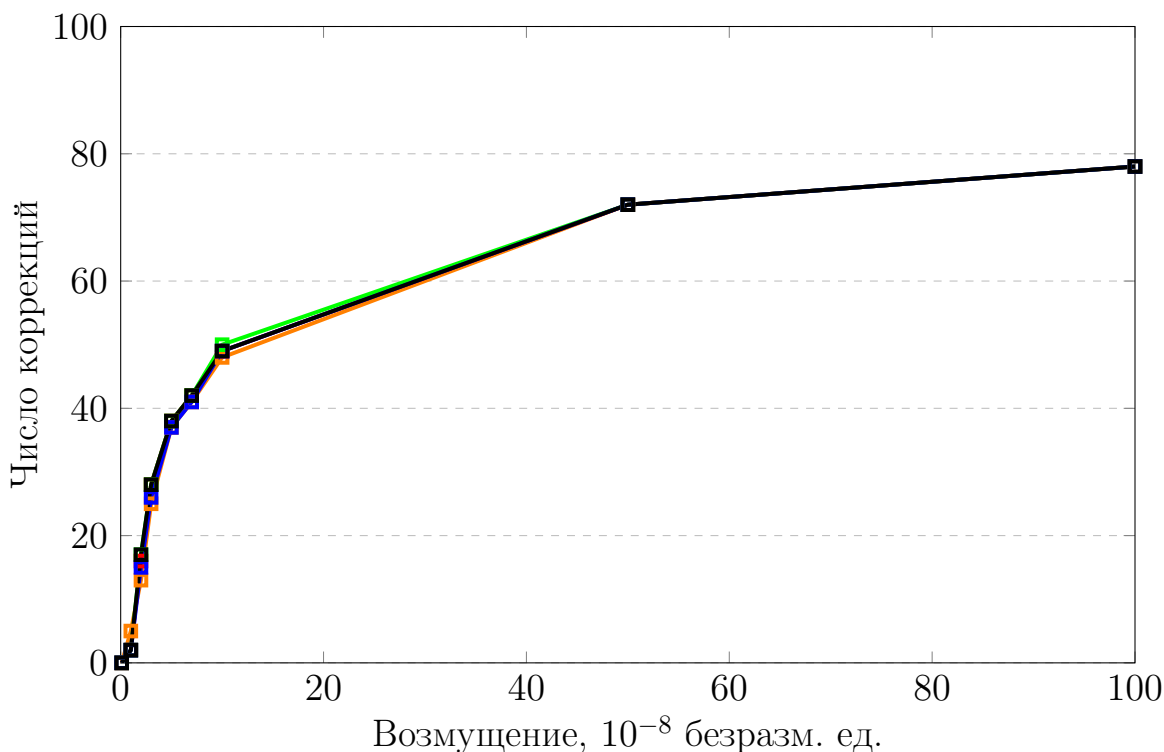


Рис. 10. Число коррекций в зависимости от силы возмущения.

По рисунку 11 можно понять, что предположение было верным, поскольку величина скачка уменьшается при увеличении количества нейронов. Черная линия, отвечающая $M = 256$ нейронам, практически его не имеет.

Осталось проверить устойчивость решения: способность нейронных сетей аппроксимировать траектории, начальные точки которых не совпадают с точкой из постановки. Для этого инициализируем набор начальных условий для радиус-векторов в шаре радиуса ε_1 с центром в точке, задаваемой начальным радиус-вектором. Начальные значения для векторов скоростей инициализируем в шаре радиуса ε_2 с центром в точке, задаваемой начальным вектором скорости. За ε_1 возьмем значение, в 10 раз превышающее значение, используемое при составлении обучающей выборки. За ε_2 тоже возьмем значение, используемое при составлении обучающей выборки, но увеличивать его не будем. В качестве проверочного функционала выберем опять Q -метрику.

Для проверки устойчивости было проделано около сотни экспериментов и во всех случаях КА попадал в 10-ти километровую окрестность конечной точки. Учитывая этот результат, можно утверждать, что решение является устойчивым.

Моделирование проводилось на персональном компьютере со следующими характеристиками: процессор Intel(R) Core I5-8520U с тактовой частотой 1.6ГГц, объем оперативной памяти 8Гб.

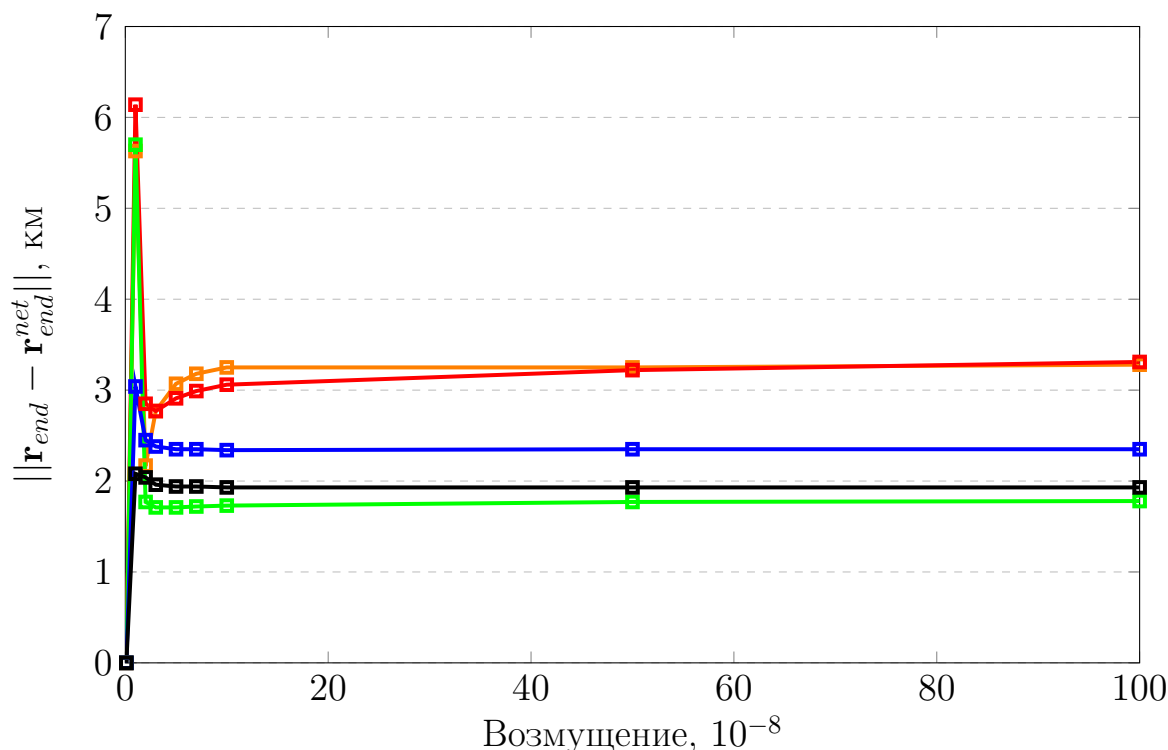


Рис. 11. Невязка по положению в зависимости от силы возмущения и M .

— $M = 16$; — $M = 32$; — $M = 64$; — $M = 128$; — $M = 256$

6. Перелет на геостационарную орбиту

Разработанная управляющая нейронная сеть была применена нами к задаче перелета с геопереходной орбиты на геостационарную орбиту. Геопереходная орбита – это орбита с высотой перицентра 250 км, высотой апоцентра 35786 км и наклоном 6° . Геостационарная орбита – это круговая орбита в геэкваториальной плоскости (наклонение 0°) высотой 35793 км. В качестве номинальной орбиты мы выбрали оптимальную траекторию перелета длительностью 35 дней (50 витков). Масса аппарата равна 3000 кг (масса типичного спутника связи). Траектория перелета вместе с геопереходной и геостационарной орбитой изображены на рисунке 12. Затраты топлива на такой перелет составляют 800 кг (26.6% от массы аппарата).

В качестве управляющей была выбрана нейронная сеть со 100 нейронами на промежуточном слое. Сеть обучалась в рамках задачи двух тел модифицированным методом Левенберга–Маркварадта с байесовской регуляризацией и обратным распространением ошибки. Так как данная траектория многовитковая, мы наблюдали высокую чувствительность к начальным условиям на со-

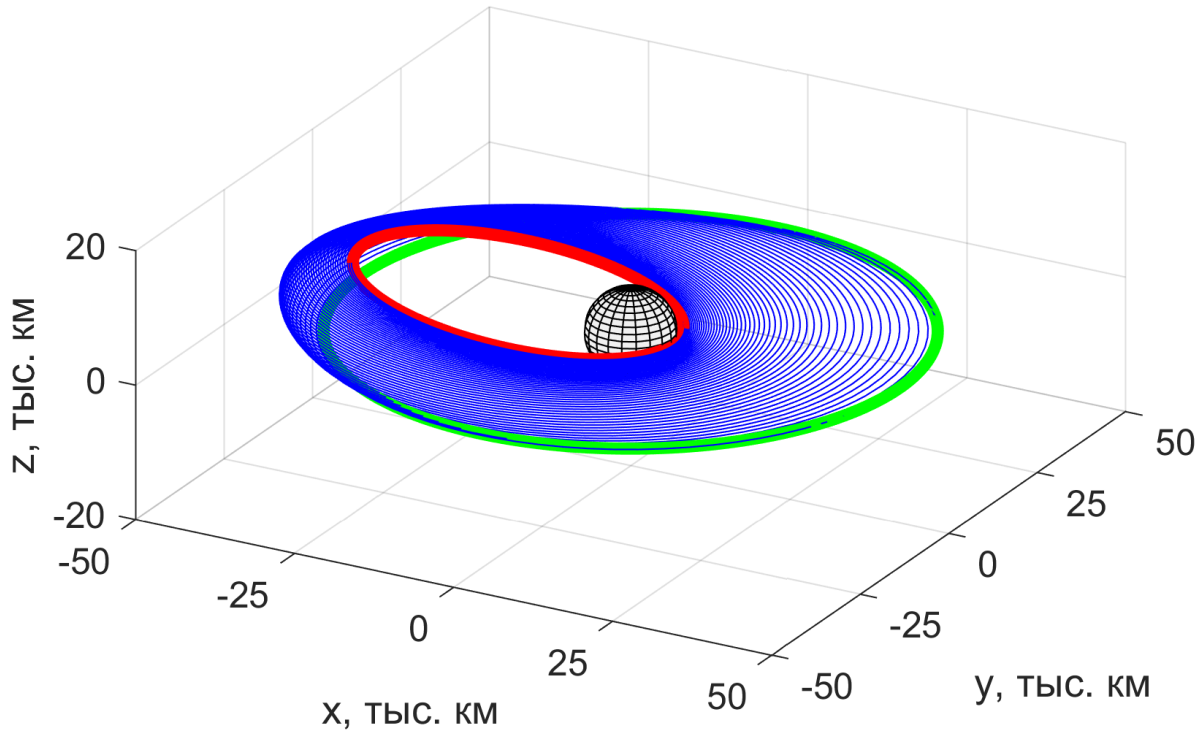


Рис. 12. Многовитковая номинальная траектория между геопереходной орбитой (красная) и геостационарной орбитой (зеленая); время полета 35 дней, всего совершается 50 витков вокруг Земли (в центре).

пряженные переменные, поэтому выборку, которая бы нацеливала аппарат на конечную точку при старте из окрестности начальной точки, построить уже не удастся. Решено было поэтому для формирования выборки нацеливаться не на конечную требуемую точку, а на точки на номинальной траектории через виток. Это решение позволило быстро сформировать выборку и обучить ее. Аппарат, двигающийся в окрестности номинальной траектории, таким образом, все время возвращает аппарат на целевую траекторию.

Работу данной нейронной сети мы протестировали и при наличии все основных возмущений, которые действуют на аппарат: силы сопротивления атмосферы, гармоник J_2 , силы светового давления, силы гравитационного притяжения Луны и силы гравитационного притяжения Солнца. Уравнения движения в такой возмущенной задаче двух тел выглядят следующим образом:

$$\frac{d^2\mathbf{r}}{dt^2} = -\frac{\mu\mathbf{r}}{r^3} + \mathbf{a}_a + \mathbf{a}_{J_2} + \mathbf{a}_{SRP} + \mathbf{a}_M + \mathbf{a}_S.$$

Здесь ускорение \mathbf{a}_S силы гравитационного притяжения Солнца вычисляется по

формуле

$$\mathbf{a}_S = \frac{\mu_S}{|\mathbf{r}_S - \mathbf{r}|^3}(\mathbf{r}_S - \mathbf{r}) - \frac{\mu_S}{|\mathbf{r}_S|^3}\mathbf{r}_S,$$

где \mathbf{r} – радиус-вектор КА, \mathbf{r}_S – радиус-вектор Солнца и $\mu_S = 1.3271244 \times 10^{11}$ км³/с² – гравитационный параметр Солнца. Такая же формула используется для расчета ускорения \mathbf{a}_M силы гравитационного притяжения Луны:

$$\mathbf{a}_M = \frac{\mu_M}{|\mathbf{r}_M - \mathbf{r}|^3}(\mathbf{r}_M - \mathbf{r}) - \frac{\mu_M}{|\mathbf{r}_M|^3}\mathbf{r}_M,$$

где \mathbf{r}_M – радиус-вектор Луны и $\mu_M = 4.9028002 \times 10^3$ км³/с² – гравитационный параметр Луны. Ускорение \mathbf{a}_{J_2} , вызванное второй зональной гармоникой геопотенциала, рассчитывается по формуле

$$\mathbf{a}_{J_2} = -\frac{3\mu_E J_2 R_E^2}{2r^4} \begin{bmatrix} \left(1 - \frac{5z^2}{r^2}\right) \frac{x}{r} \\ \left(1 - \frac{5z^2}{r^2}\right) \frac{y}{r} \\ \left(3 - \frac{5z^2}{r^2}\right) \frac{z}{r} \end{bmatrix},$$

где $r = |\mathbf{r}|$, $J_2 = 1.0826 \times 10^{-3}$, $R_E = 6378.13$ км – экваториальный радиус Земли. Ускорение \mathbf{a}_A , вызванное силой сопротивления атмосферы Земли, равно

$$\mathbf{a}_A = -f \frac{\mathbf{v}}{v},$$

где $v = |\mathbf{v}|$ – скорость КА и

$$f = \frac{c_x A v^2}{2M} \rho_l \left(\frac{\rho_h}{\rho_l} \right)^{(D-D_0)/(D_1-D_0)} \quad (2)$$

Здесь $c_x = 2.2$ – коэффициент лобового сопротивления, $A = 60$ м² – характерная площадь поперечного сечения КА. Формула (2) представляет кусочно-экспоненциальную модель атмосферы и дает ускорение на высоте D для выбранной пары плотностей ρ_l и ρ_h на высотах D_0 и D_1 , соответственно; $D_0 \leq D \leq D_1$. Величины для плотностей, используемые в данной работе, соответствуют высотным профилям модели JB2008 и могут быть найдены в COSPAR International Reference Atmosphere 2012. Для высоты больше 900 км полагается $f = 0$.

Траектория аппарата, двигающегося по тому же закону оптимального управления, что и в невозмущенной модели, в рамках возмущенной модели изображена на рисунке 13. В результате воздействия возмущений, параметры орбиты в

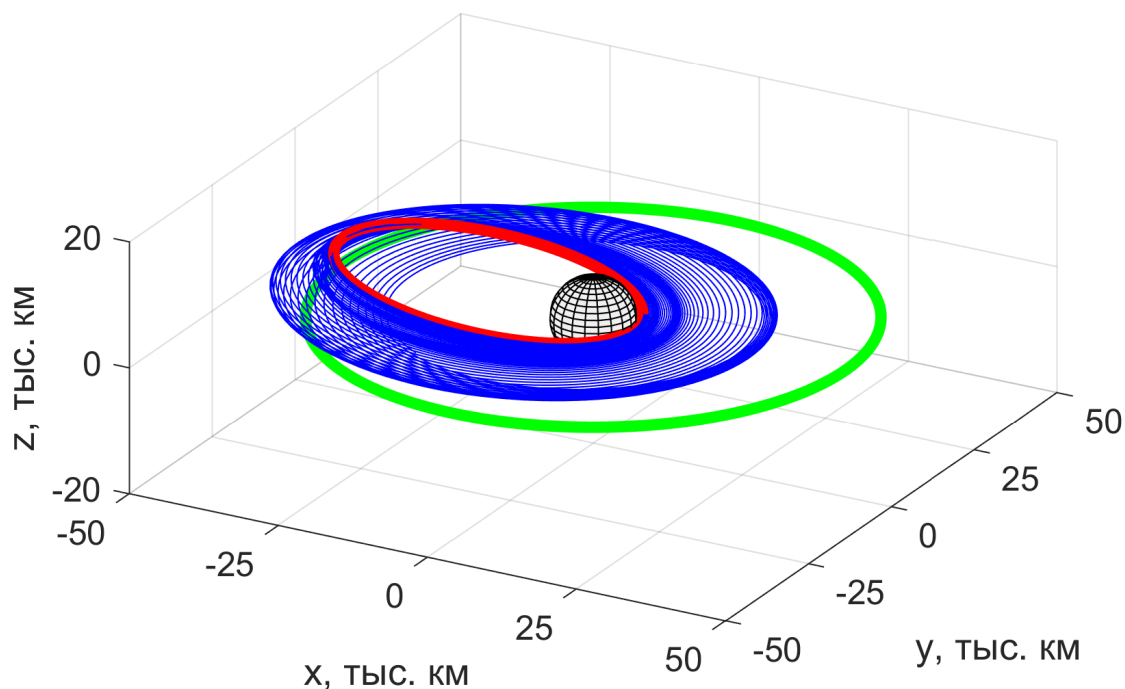


Рис. 13. Многовитковая возмущенная номинальная траектория между геопереходной орбитой (красная) и геостационарной орбитой (зеленая); время полета 35 дней, всего совершается 50 витков вокруг Земли (в центре).

конце полета сильно отличаются от требуемых. Так, например, большая полуось орбиты равна 27681 км вместо требуемой 42164 км для геостационарной орбиты, эксцентриситет равен 0.1, а наклонение 0.8° . Все это в очередной раз говорит о неадекватности использования программного управления в возмущенной среде и подчеркивает необходимость использования систем с обратной связью и адаптивных систем управления.

После встраивания обученной в простой модели нейронной сети, мы получаем управление с обратной связью. В каждый момент времени это управление нацеливает аппарат на кривую, отвечающей номинальному движению в невозмущенной задаче. Из-за наличия возмущений, аппарат больше не движется по номинальной траектории, однако тяги хватает для того, чтобы поддерживать движения вблизи номинальной траектории. Большая полуось фактической орбиты отличается от большой полуоси геостационарной орбиты на 516.8

км, эксцентриситет фактической орбиты равен 0.0127, а наклонение 0.19° . Отметим также высокую невязку по аргументу широты величиной 47° . Эта невязка устраняется коррекцией времени полета. В нашем случае достаточно лететь примерно на 0.12 дней дольше. Невязка по орбитальным переменным в этом случае практически не изменится: по большой полуоси она станет 596.9 км, по эксцентриситету — 0.0114, а по наклонению — 0.16° .

Отличия между реактивным ускорением, выдаваемом нейронной сетью в возмущенной модели, и ускорением на номинальной траектории в невозмущенной модели, проявляются ближе к концу траектории, см. рисунок 14.

Чтобы оценить, сколько уходит затрат топлива на такое поддержание, мы вычли из массы $m_f(t)$ аппарата, двигающегося с использованием нейронной сети в возмущенной модели, массу $m_n(t)$ аппарата, двигающегося согласно номинальному движению в невозмущенной задаче. Зависимость $m_f(t) - m_n(t)$ от времени полета t изображена на рисунке 15. Хорошо видно, итоговые затраты на поддержание составляют порядка 8 кг, что меньше 0.3% от массы аппарата и 1% от затрат топлива на перелет.

В настоящей работе, помимо создания нейронной сети, способной поддерживать движение аппарата, также рассматривалась возможность «дообучения» нейронной сети в полете, т.е. возможности коррекции ее весов таким образом, чтобы нивелировать невязку по положению и скорости в конце полета. Наиболее простым решением здесь оказалось проварьировать каждый из весов, составить матрицу Якоби невязки по фазовому состоянию в конце полета и выполнить один шаг в методе Ньютона минимальной нормы. Так как чувствительность к весам сети в начале пути слишком высокая, такую коррекцию разумно осуществлять в конце траектории, за 4–5 витков до конца. Отметим, что для оценки матрицы Якоби по разностным схемам уравнения движения интегрировать не приходится: оценку вектора состояния в конце траектории можно произвести с помощью заранее обученной прогнозирующей сети. Несмотря на то, что такая сеть обучается в рамках невозмущенной задачи двух тел, возмущения не успевают достаточно сильно возмутить траекторию за малое время до конца движения.

Схема коррекции весов управляющей сети получается следующей. За 5 витков до конца движения с помощью прогнозирующей сети оценивается невязка по фазовому состоянию в конце движения $\Delta \mathbf{x}$. Далее, каждый из параметров θ_i управляющей сети по отдельности варьируется на некоторую малую величину $\Delta \theta_i$. Для каждой такой вариации управляющая сеть выдает вектор сопряженных переменных, которые должны привести в целевую точку в конце траектории. Прогнозирующая нейронная сеть принимает данный вектор сопряженных переменных и оценивает невязку по фазовому состоянию в конце движения. Так

формируется матрица производных J невязки по каждому параметру управляющей нейронной сети. Далее рассчитывается шаг минимальной нормы метода Ньютона для коррекции параметров θ :

$$\Delta\theta = -J^T (JJ^T + \lambda E)^{-1} \Delta\mathbf{x},$$

где λ – вспомогательный параметр, играющий такую же роль, как в методе Левенберга–Марквардта. После того, как коррекция принята, аппарат движется дальше до следующего витка, где описанная процедура повторяется вновь. Чем ближе аппарат находится к цели, тем лучше оценивает прогнозирующая сеть конечное положение и скорость аппарата и тем точнее выполняется коррекция параметров управляющей сети. Отметим также, что коррекции подвергается и конечный момент времени t_f . Производная невязки по t_f вычисляется аналитически: она равна вектору правых частей уравнения движения в этот момент времени. Так как реальная модель движения аппарату неизвестна, этот вектор рассчитывается исходя из модели задачи двух тел.

Такой подход позволил нам понизить промах по положению и скорости. Так невязка по большой полуоси уменьшилась почти в 10 раз до 60.1 км, эксцентриситет уменьшился до 0.0005, а наклонение до 0.02° . Все это показывает принципиальную возможность уменьшать невязку за счет коррекции параметров управляющей нейронной сети. Дальнейшие исследования могут быть направлены на полное устранение невязки, хотя с практической точки зрения в этом может не быть необходимости из-за наличия навигационных ошибок и ошибок исполнения реактивной тяги.

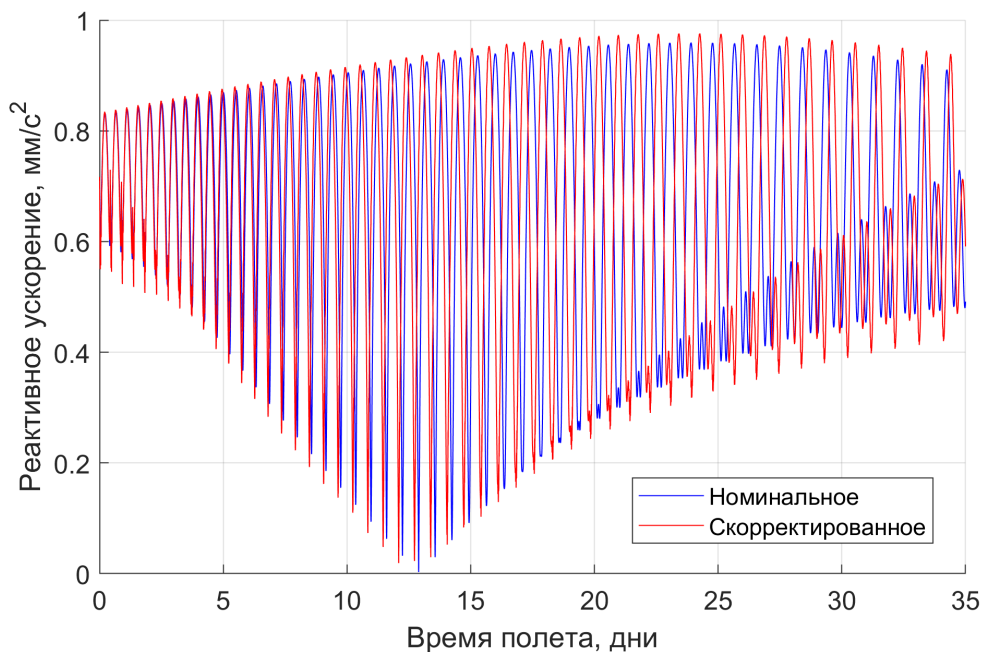


Рис. 14. Реактивное ускорение при движении по номинальной траектории в невозмущенной модели (синяя кривая) и реактивное ускорение, выдаваемое нейронной сетью в возмущенной модели.

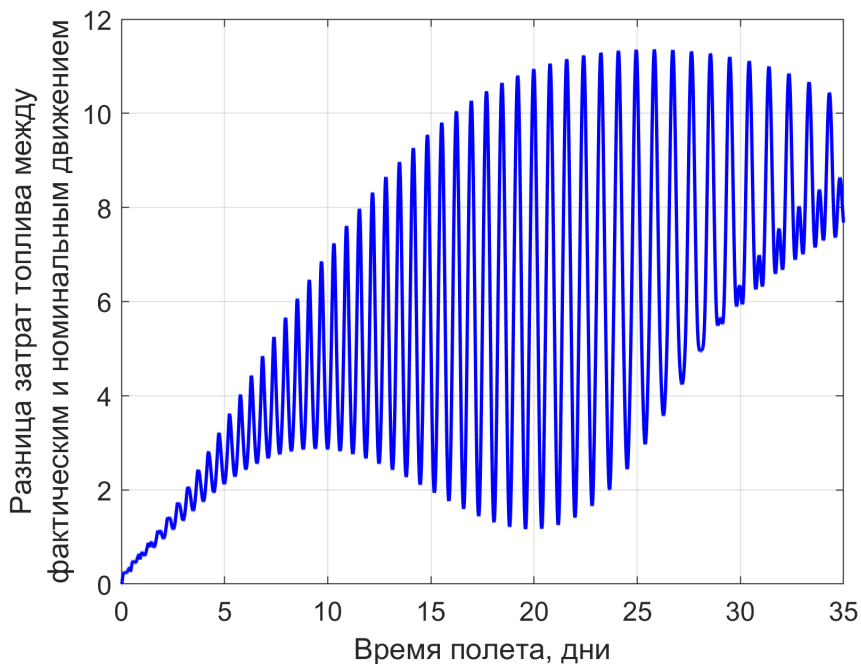


Рис. 15. Разница между массой аппарата с нейронной сетью в возмущенной модели и массой аппарата на номинальной траектории в невозмущенной модели в зависимости от времени полета.

7. Заключение

Разработаны две искусственные нейронные сети (управляющая и прогнозирующая), которые могут быть использованы для коррекции орбитального движения КА вблизи спиралевидной номинальной траектории. Как управляющая так и прогнозирующая нейронные сети имеют один скрытый слой. Для управляющей сети выборка формировалась в результате решения совокупности краевых задач принципа максимума Понтрягина методом Левенберга–Марквардта в окрестности номинального решения. Для прогнозирующей сети выборка формируется в результате решения множества задач Коши варьированием фазового вектора в пространстве прямых и сопряженных переменных. Обучение нейронных сетей проводилось в рамках невозмущенной задачи двух тел. Тестирование нейронных сетей в возмущенной задаче двух тел показало: 1) найден порог возмущений, при котором не требуется корректирование движения (аппарат попадает в целевую точку с нужной точностью), 2) с ростом уровня возмущений увеличивается число необходимых коррекций, 3) поведение нейронных сетей оказалось устойчивым к расширению области начальных условий (свойство обобщения нейронных сетей). Кроме того, рассмотрена модельная задача перелета между геопереходной и геостационарной орбитами в рамках возмущенной задачи двух тел, учитывающей возмущения от сил атмосферного сопротивления, гармоник J_2 , сил светового давления, а также лунно-солнечных гравитационных взаимодействий. Было показано, что управляющая нейронная сеть успешно поддерживает движение вблизи номинальной траектории, а затраты на поддержание не превышают процента от затрат топлива для номинального движения. Описан и способ адаптации параметров нейронной сети для того, чтобы уменьшить невязку по фазовому состоянию в конце полета. Результаты показали принципиальную возможность уменьшения невязки. Дальнейшие исследования могут быть направлены на ее полное устранение.

8. Список литературы

- [1] KrishnaKumar K., Rickard S., Bartholomew S. Adaptive neuro-control for spacecraft attitude control // *Neurocomputing*. 1995. Vol. 9, no. 2. P. 131–148.
- [2] Artificial neural networks in Space Station optimal attitude control / R. R. Kumar, H. Seywald, S. M. Deshpande et al. // *Acta Astronautica*. 1995. Vol. 35, no. 2-3. P. 107–117.
- [3] Dachwald B. Optimization of very-low-thrust trajectories using evolutionary neurocontrol // *Acta Astronautica*. 2005. Vol. 57, no. 2-8. P. 175–185.
- [4] Carnelli I., Dachwald B., Vasile M. Optimizing low-thrust gravity assist interplanetary trajectories using evolutionary neurocontrollers // *2007 IEEE Congress on Evolutionary Computation / IEEE*. 2007. P. 965–972.
- [5] Carnelli I., Dachwald B., Vasile M. Evolutionary neurocontrol: A novel method for low-thrust gravity-assist trajectory optimization // *Journal of guidance, control, and dynamics*. 2009. Vol. 32, no. 2. P. 616–625.
- [6] Ohndorf A., Dachwald B., Gill E. Optimization of low-thrust earth-moon transfers using evolutionary neurocontrol // *IEEE Congress on Evolutionary Computation, 2009. CEC'09. / IEEE*. 2009. P. 358–364.
- [7] Ohndorf A. Multiphase Low-Thrust Trajectory Optimization Using Evolutionary Neurocontrol. PhD. Thesis. Delft University of Technology, Delft, Netherlands, 2016.
- [8] Gurfil P., Idan M., Kasdin N. J. Adaptive neural control of deep-space formation flying // *Journal of guidance, control, and dynamics*. 2003. Vol. 26, no. 3. P. 491–501.
- [9] Bae J., Kim Y. Adaptive controller design for spacecraft formation flying using sliding mode controller and neural networks // *Journal of the Franklin Institute*. 2012. Vol. 349, no. 2. P. 578–603.
- [10] Izzo D., Simões L. F., de Croon G. C. An evolutionary robotics approach for the distributed control of satellite formations // *Evolutionary Intelligence*. 2014. Vol. 7, no. 2. P. 107–118.
- [11] Pérez D. Adaptive lyapunov control and artificial neural networks for spacecraft relative maneuvering using atmospheric differential drag. PhD. Thesis: Rensselaer Polytechnic Institute. 2013.

- [12] Orbit-centered atmospheric density prediction using artificial neural networks / D. Pérez, B. Wohlberg, T. A. Lovell et al. // *Acta Astronautica*. 2014. Vol. 98. P. 9–23.
- [13] Pérez D., Bevilacqua R. Neural Network based calibration of atmospheric density models // *Acta Astronautica*. 2015. Vol. 110. P. 58–76.
- [14] Modelling irregular small bodies gravity field via extreme learning machines / R. Furfaro, R. Linares, V. Reddy et al. // 27th AAS/AIAA Space Flight Mechanics Meeting, 2017 / Univelt Inc. 2017. P. 471–486.
- [15] Жданов А. А., Земских Л. В., Беляев Б. Б. Система стабилизации углового движения космического аппарата на основе нейроноподобной системы автономного адаптивного управления // *Космические исследования*. 2004. Т. 42, № 3. С. 1–15.
- [16] Жданов А. А. Автономный искусственный интеллект. М.: БИНОМ. Лаборатория знаний, 2014.
- [17] Шишкин О. Г. Искусственная нейронная сеть для диагностики космического аппарата // *Интеллектуальные системы, управление и мехатроника-2016*. 2016. С. 448–451.
- [18] Воронцов В. А., Федоров Е. А. Разработка прототипа интеллектуальной системы оперативного мониторинга и технического состояния основных бортовых систем космического аппарата // *Труды МАИ*. 2015. № 82. С. 35.
- [19] Косинский М. Ю., Шатский М. А. Разработка моделей и методики для анализа и прогнозирования надёжности бортовых систем управления космических аппаратов на основе теории нечётких множеств и искусственных нейронных сетей // *Труды МАИ*. 2014. № 74. С. 23.
- [20] Нейросетевая система контроля датчиков углов ориентации и дальности космического аппарата / Ю. Г. Емельянова, К. А. Константинов, С. В. Погодин [и др.] // *Программные системы: теория и приложения*. 2010. Т. 1, № 1.
- [21] Петухов В.Г. Оптимизация межпланетных траекторий космических аппаратов с идеально-регулируемым двигателем методом продолжения // *Космические исследования*. 2008. Т. 46, № 3. С. 224–237.

Оглавление

1. Введение	3
2. Постановка задачи	4
3. Введение в нейронные сети	6
4. Решение задачи перелета с малой тягой	9
5. Моделирование	14
6. Перелет на геостационарную орбиту	21
7. Заключение	28
8. Список литературы	29