

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ
(национальный исследовательский университет)

ФИЗТЕХ-ШКОЛА ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ
Кафедра математического моделирования и прикладной математики

Квалификационная работа на соискание степени магистра
по направлению 03.04.01 «Прикладные математика и физика»,
магистерская программа «Управление динамическими системами»

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

Выполнил:

студент группы 872

Сорокин Артемий

Научный руководитель:

к.ф.-м.н., доцент

Широбоков Максим Геннадьевич

Москва, 2020

Аннотация

Работа посвящена применению искусственных нейронных сетей в задаче оптимального по энергии перелета с двигателями малой тяги. Рассматривается перелет с околоземной орбиты на геостационарную орбиту в рамках возмущенной модели задачи двух тел. Предполагается, что бортовой компьютер космического аппарата оснащен искусственной нейронной сетью, отвечающей за управление движением, и нейронной сетью, прогнозирующей время его полета. В докладе описывается реализация предлагаемых нейронных сетей, обосновывается выбор архитектуры, а также приводятся результаты исследования минимального размера нейронных сетей, удовлетворяющих требованиям к отказоустойчивости и устойчивости к моделируемым возмущениям и во время полета.

Оглавление

Введение	4
1. Цель работы	5
2. Постановка задачи	6
3. Введение в нейронные сети	8
4. Решение задачи	13
4.1. Перелет с малой тягой	13
4.2. Составление обучающих выборок	15
4.3. Архитектура ИНС	18
5. Моделирование	21
Заключение	30
Список использованных источников	31

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. В последнее время также 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 [3–7]. 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 [8–10]. Как было показано, нейронные сети способны парировать случайные возмущения и неточность модели даже в условиях неустойчивой динамики вокруг коллинеарных точек либра-

ции [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]).

Отметим, наконец, что значительная часть отечественных работ в области использования нейронных сетей в космической деятельности сосредоточена на средствах диагностики состояния аппаратуры на спутнике (например, [11–14]).

1. Цель работы

Цель работы состоит в разработке нейронных сетей для управления движением космического аппарата в режиме реального полета. Поскольку к космической технике предъявляются определенные требования на производительность и надежность, то данное исследование направлено на решение проблемы поиска минимально возможных нейронных сетей, отвечающих требованиям к точности и устойчивости к моделируемым возмущениям.

Поскольку в реальном полете высокоэнергетические частицы непрерывно бомбардируют космический аппарат и его бортовой компьютер, они способны нарушить работу блока управления. Таким образом в исследовании также поднимается проблема проверки отказоустойчивости полученного решения.

В качестве примера был выбран перелет с околоземной орбиты на геостационарную орбиту с двигателями малой тяги.

2. Постановка задачи

Рассмотрим орбитальное движение космического аппарата (КА). Аппарат снабжен двигателем малой тяги и движется в центральном гравитационном поле Земли согласно уравнениям движения:

$$\begin{cases} \dot{\mathbf{r}} = \mathbf{v}, \\ \dot{\mathbf{v}} = -\frac{\mu\mathbf{r}}{r^3} + \mathbf{u}, \end{cases} \quad (1)$$

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.

Введем вспомогательные предположения, переменные и уравнения, чтобы поставить задачу. Пусть требуется перевести аппарат из точки \mathbf{r}_0 , где он имеет скорость \mathbf{v}_0 , в точку \mathbf{r}_1 , где он должен иметь скорость \mathbf{v}_1 , за время T . Среди всевозможных функций управления \mathbf{u} нас будет интересовать та, которая доставляет минимум функционалу

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

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

$$\begin{cases} \dot{\mathbf{r}} = \mathbf{v}, \\ \dot{\mathbf{v}} = -\frac{\mu\mathbf{r}}{r^3} + \boldsymbol{\lambda}_v, \\ \dot{\boldsymbol{\lambda}}_r = \frac{\mu\boldsymbol{\lambda}_v}{r^3} - \frac{3\mu\mathbf{r}^T\boldsymbol{\lambda}_v}{r^5}\mathbf{r}, \\ \dot{\boldsymbol{\lambda}}_v = -\boldsymbol{\lambda}_r. \end{cases} \quad (2)$$

Будем считать управление $\mathbf{u}^*(t)$ оптимальным, если оно удовлетворяет необходимым условиям принципа максимума Понтрягина, т.е. начальным условиям $\mathbf{r}(0) = \mathbf{r}_0$, $\mathbf{v}(0) = \mathbf{v}_0$ и краевым условиям $\mathbf{r}(T) = \mathbf{r}_1$, $\mathbf{v}(T) = \mathbf{v}_1$. Это управление определяется неизвестными начальными условиями $\boldsymbol{\lambda}_r(0)$, $\boldsymbol{\lambda}_v(0)$, которые при интегрировании уравнений (2) вместе с начальными условиями $\mathbf{r}(0) = \mathbf{r}_0$, $\mathbf{v}(0) = \mathbf{v}_0$ приводят к равенствам $\mathbf{r}(T) = \mathbf{r}_1$, $\mathbf{v}(T) = \mathbf{v}_1$. Отметим, что функция управления $\mathbf{u}^*(t) = \boldsymbol{\lambda}_v(t)$ определяется из решения уравнений (2) 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, [15]). Число витков N_{rev} вокруг Земли при этом может быть выбрано произвольно.

Итак, пусть для конкретных \mathbf{r}_0 , \mathbf{v}_0 , \mathbf{r}_1 , \mathbf{v}_1 , T и N_{rev} было найдено оптимальное управление $\mathbf{u}^*(t)$, которое мы назовем номинальным управлением, а траекторию, которую оно определяет, номинальной траекторией. Целью данной работы является построение набора нейросетевых регуляторов для поддержания КА в окрестности номинальной траектории в условиях реального полета, а также поиск минимального размера данных регуляторов, при котором они будут удовлетворять требованиям к точности попадания в \mathbf{r}_1 , отказоустойчивости и устойчивости к моделируемым возмущениям.

В качестве одного из выше упомянутых регуляторов будет выступать ИНС, аппроксимирующая отображение $\mathbf{u}_{ctrl} : (\mathbf{r}(t), \mathbf{v}(t)) \rightarrow \boldsymbol{\lambda}_v(t)$. Дан-

ное отображение сопоставляет фазовому вектору значение реактивного ускорения, необходимого для поддержания КА вблизи номинальной траектории. Эту нейронную сеть мы называем *управляющей* нейронной сетью. В качестве второго регулятора выступит ИНС, аппроксимирующая отображение $(\mathbf{r}(t), \mathbf{v}(t), \boldsymbol{\lambda}_v(t)) \rightarrow T_{left}$, вблизи номинальной траектории. Данное отображение сопоставляет фазовому вектору и вектору сопряженных к скорости переменных оставшееся время полета КА до конечной точки \mathbf{r}_1 . Эту нейронную сеть мы назовем *прогнозирующей* нейронной сетью.

В последующих главах работы приводится описание архитектуры разрабатываемых нейронных сетей, метод формирования обучающей выборки, метод обучения, а также некоторые результаты, касательно применения данных нейронных сетей к задаче управления аппаратом при многовитковом перелете между геопереходной и геостационарной орбитами. В следующей главе дадим краткое определение нейронной сети, а также рассмотрим метод изменения весов в ИНС, называемый обучением с учителем.

3. Введение в нейронные сети

На текущий момент нейронные сети занимают лидирующие позиции в области решений многих прикладных задач, таких как распознавание изображений, текста, речи, синтеза речи и других. Все эти задачи принадлежат трем следующим классам: задача классификации, кластеризации и регрессии объектов. Будем рассматривать только последний класс задач и соответствующие методики, т.к. в работе потребуется решать именно задачу регрессии. Под регрессией будем понимать поиск такого отображения f , которое описывает влияние одной или нескольких независимых переменных X_1, X_2, \dots, X_p на зависимую переменную Y . В качестве универсального аппроксиматора данного отображения f могут выступать нейронные сети.

Дадим формальное определение ИНС: нейронная сеть – это взве-

шенный ориентированный граф, в узлах которого находятся функции, называемые активационными. Активационная функция представляет из себя некоторую монотонную и непрерывно дифференцируемую функцию. Процесс изменения весов в графе называется ее обучением. С математической точки зрения обучение нейронных сетей — это многопараметрическая задача нелинейной оптимизации. Существует несколько способов обучения ИНС: обучение с учителем, обучение без учителя и обучение с подкреплением, однако для задачи регрессии применяется именно обучение с учителем. В данной методике для настройки параметров используется массив данных, называемый обучающей выборкой, состоящий из входных сигналов и соответствующих им выходных сигналов. В данном разделе коротко рассматриваются метод обучения с учителем, соответствующие нейронные сети и их архитектуры.

Определим структуру графа ИНС: выделим 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.

В каждом узле графа размещена активационная функция единственного аргумента. Приведем примеры часто используемых активационных функций при решении задачи регрессии. Начнем с тождественной: $\phi(x) = x$. Она является дифференцируемой и монотонной, однако ее производная постоянна, а значит градиент является постоянным вектором и спуск производится по постоянному градиенту. Таким образом если производится неточное предсказание, то тогда изменения, сделанные при обратным рас-

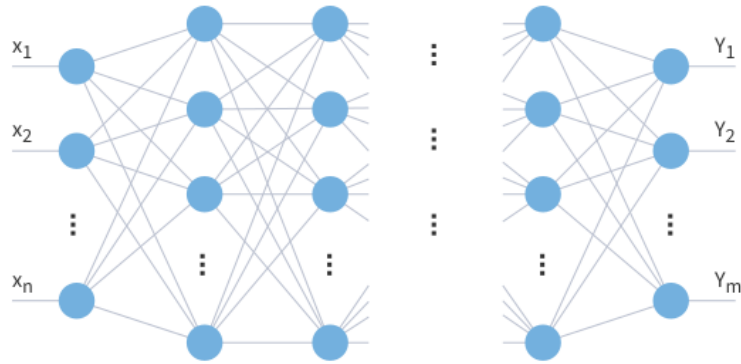


Рисунок 1 – Многослойная нейронная сеть.

пространении ошибки, тоже будут постоянными и не будут зависеть от изменений на входе. В случае если все активационные функции в сети линейные - ее выход также будет линейным, а именно являться суперпозицией линейных функций активаций. Таким образом аппроксимировать нелинейное отображение не удастся. Следующая активационная функция – сигмоида: $\phi(x) = 1/(1 + e^{-x})$. Она нелинейная и комбинация таких функций производит тоже нелинейную функцию. Для нее также характерен гладкий градиент. Однако при приближении к краям сигмоиды значения $\phi(x)$ начинают слабее меняться в зависимости от x . Градиент в таких областях начинает принимать маленькие по модулю значения, а это, в свою очередь, приводит к тому, что нейронная сеть перестает обучаться на примерах с большими значениями. Рассмотрим еще одну функцию – гиперболический тангенс: $\phi(x) = \text{th}(x)$. Гиперболический тангенс можно выразить через сигмоиду, поэтому он имеет те же особенности. Однако диапазон значений функции лежит в интервале $(-1, 1)$, а производная растет быстрее, чем у сигмоиды.

При построении ИНС активационная функция должна быть одинаковой для всех нейронов в слое, но может различаться для разных слоев. 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.

Приведем примеры известных функционалов, используемых при решении задачи регрессии. Для большинства оптимизационных алгоритмов предпочтительно иметь дифференцируемую функцию потерь, но это требование не обязательно. Рассмотрим, к примеру, средний модуль отклонения:

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

где y_i – i -я компонента выходящего сигнала, \tilde{y}_i – i -я компонента предсказанного сигнала, а N – количество компонент выходного сигнала. Проблема данного функционала заключается в том, что он не дифференцируем в точках, когда y_i совпадает с \tilde{y}_i , а оптимизационные алгоритмы склонны застревать в этих точках. Поэтому рассмотрим еще один часто используемый функционал – среднеквадратичное отклонение:

$$E_{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \tilde{y}_i)^2.$$

Преимущество по сравнению с E_{MAE} заключается в дифференцируемости, однако недостаток кроется в том, что функционал является чувствительным

к выбросам. Это можно объяснить тем, что чем сильнее значение выходного сигнала будет отличаться от предсказанного, тем больший вклад оно внесет.

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. В данной работе для обновления весов в сетях используется метод вычисления градиента, называемый методом обратного распространения ошибки. Рассмотрим принцип работы этого алгоритма на примере простейшего функционала ошибки, приведенного ранее — $E_{MSE} = \frac{1}{N} \sum_{i=1}^N (d_i - t_i)^2$, 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: $v_j^l = \sum_{i=0}^M w_{ji}^l y_i^{l-1}$, 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 $l - 1$. Выходной сигнал нейрона j слоя l после применения активационной функции ϕ 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 $e_j = (d_j - t_j)^2/N$. 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 является итерационным и завершается в том случае, когда функционал ошибки достигает определенного значения.

4. Решение задачи

4.1. Перелет с малой тягой

Рассматривается задача многовиткового перелета между геопереходной и геостационарной орбитами. Геопереходная орбита – это орбита с высотой перицентра 250 км, высотой апоцентра 35786 км и наклоном 6° . Геостационарная орбита – это круговая орбита в экваториальной 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 2. Затраты топлива на такой перелет составляют 800 кг (26.6% от массы аппарата).

Зафиксировав начальные и конечные условия, необходимо свести краевую задачу к задаче Коши. Сведение производится методом продолжения по гравитационному параметру, описанному в [16]. Реализованный в среде *MATLAB*, данный метод носит название *EnergyOptimalProblem*. В качестве аргументов метод принимает значения начального и конечного

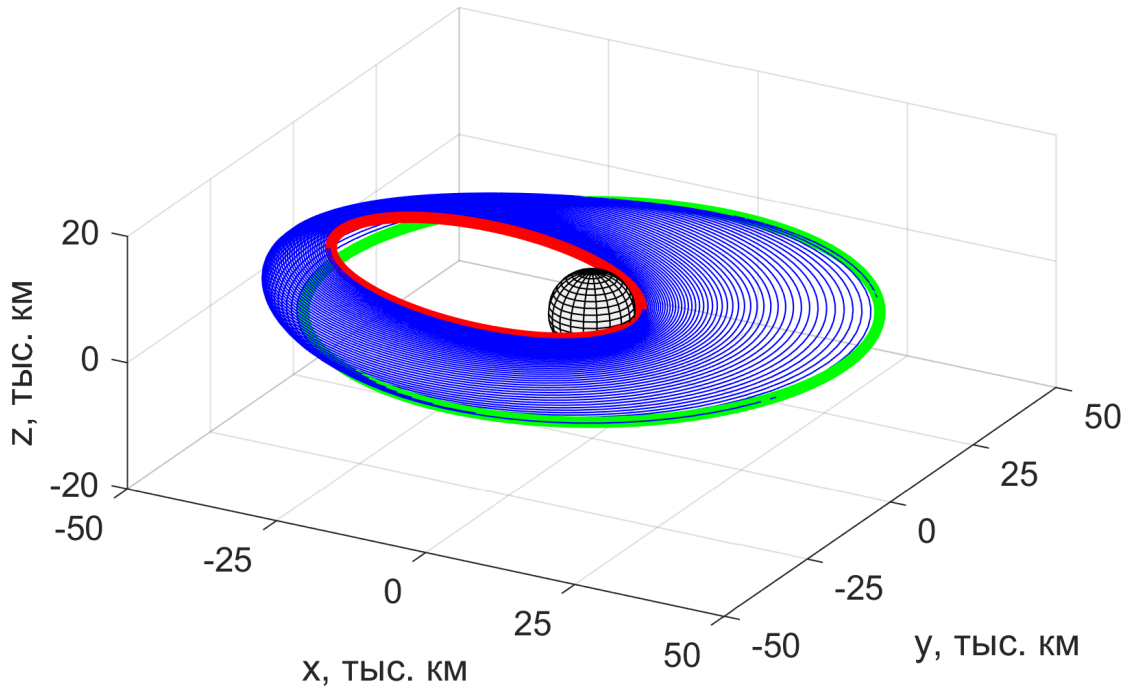


Рисунок 2 – Многовитковая номинальная траектория между геопереходной орбитой (красная) и геостационарной орбитой (зеленая); время полета 35 дней, всего совершается 50 витков вокруг Земли (в центре).

фазовых векторов $\mathbf{x}_0 = [\mathbf{r}_0; \mathbf{v}_0]$ и $\mathbf{x}_1 = [\mathbf{r}_1; \mathbf{v}_1]$, время, выделенное для перелета T , и количество витков N_{revs} относительно притягивающего центра. Результатом его работы является вектор сопряженных переменных $\boldsymbol{\lambda}_0$ в начальный момент времени. Полученный вектор сопряженных переменных в совокупности с начальным значением фазового вектора дает нам начальное условие задачи Коши для оптимального перелета $\mathbf{z}_0 = [\mathbf{x}_0; \boldsymbol{\lambda}_0]$. Интегрирование будем производить методом Рунге–Кутты 8-го порядка с постоянным шагом.

Важно понимать, что расчет оптимальной траектории происходит в невозмущенной модели, т.е. в правой части дифференциального уравнения для $\dot{\mathbf{v}}$ нет возмущающего ускорения и интегрируются 12 уравнений (2). В условиях реального полета знание о сопряженных переменных будет отсутствовать, и дифференциальная система будет состоять из 6-ти уравнений (1). Таким образом возникает потребность построения некоторого регулятора, задающего управление, который должен поддерживать орбиту

вблизи оптимальной, а также парировать возмущающие ускорения.

Построим данный регулятор с помощью нейронных сетей. В постановке задачи упоминались две ИНС - *управляющая* и *прогнозирующая*. Теперь можно более подробно рассмотреть их назначения. *Управляющая* нейронная сеть по своей сути есть аппроксимация оптимального управления, т.е. для любого фазового вектора вблизи номинальной траектории ее выходом будет являться значение реактивного ускорения \mathbf{u}_{ctrl} в этой точке, необходимого для попадания в поставленную цель. Поскольку данное управление больше не будет являться оптимальным, то в процессе полета нужно будет уточнять оставшееся время до попадания в цель. Для решения данной задачи вводится *прогнозирующая* сеть, уточняющая время полета на основе текущего фазового вектора и значения реактивного ускорения.

4.2. Составление обучающих выборок

Перейдем к составлению обучающих выборок. Начнем с выборки для управляющей сети: для начала необходимо определить ограниченную область фазового пространства, в которой она будет составлена. В качестве этой области будем брать некоторую трубку траекторий относительно номинальной. Построение выборки заключается в следующих шагах: вариация начального фазового вектора $\mathbf{x}'_0 = [\mathbf{r}_0 + \boldsymbol{\varepsilon}_1; \mathbf{v}_0 + \boldsymbol{\varepsilon}_2]$, расчет сопряженных переменных $\boldsymbol{\lambda}'_0$ для \mathbf{x}'_0 и \mathbf{x}_1 , интегрирование траектории с рассчитанным начальным условием $\mathbf{z}'_0 = [\mathbf{x}'_0; \boldsymbol{\lambda}'_0]$. Получив множество точек, задающих траекторию, добавляем их в выборку и повторяем процедуру заданное количество раз.

Теперь определим, как производить вариацию начальных условий таким образом, чтобы траектории в трубке были относительно близки к номинальной траектории. Для этого введем метрику расхождения двух траекторий — сумму норм разностей фазовых векторов номиналь-

ной и проварьированной траекторий для каждой точки интегрирования $S_d = \sum_{i=1}^N \|\mathbf{x}_{nom}^i - \mathbf{x}_{var}^i\|$, где N - количество шагов интегрирования. Данная метрика не имеет физического смысла, а только показывает порядок расхождения траекторий. Стоит заметить, что выбор постоянного шага при интегрировании обусловлен именно метрикой S_d , т.к. расхождение по времени в случае адаптивного шага будет лишать ее смысла.

Проварьлируем набор траекторий для $\boldsymbol{\varepsilon}_1 = \mathbf{0}$ и произвольного малого $\boldsymbol{\varepsilon}_2$, и зафиксируем гистограмму для метрики S_d . Теперь сделаем то же самое для произвольного малого $\boldsymbol{\varepsilon}_1$ и $\boldsymbol{\varepsilon}_2 = \mathbf{0}$. Численный эксперимент показывает, что гистограммы метрики S_d совпадают когда $\|\boldsymbol{\varepsilon}_1\| \sim 2\|\boldsymbol{\varepsilon}_2\|$, т.е. окрестность для вектора скорости должна быть в два раза меньше. Осталось определить значение нормы любого из векторов $\boldsymbol{\varepsilon}_1$ или $\boldsymbol{\varepsilon}_2$. Для этого введем дополнительное дифференциальное уравнение в систему уравнений (2) для массы КА: $\dot{m} = -F/v_{ex}$, где v_{ex} - скорость истечения газов из двигателя. Поскольку мы рассматриваем оптимальный по энергии перелет с малой тягой, то сила тяги $\mathbf{F} = m\mathbf{u} = m\boldsymbol{\lambda}_v$. Также будем считать, что мощность двигателя фиксирована, а это равносильно тому, что фиксировано произведение силы тяги на скорость истечения, т.е. $Fv_{ex} = const$. Тогда перепишем уравнение для массы в следующем виде: $\dot{m} = -(m\lambda_v)^2/Fv_{ex}$.

Теперь введем еще одну метрику $S_m = |(m_{nom} - m_{var})/m_{nom}|$, где m_{nom} и m_{var} - массы КА в конечной точке для номинальной и проварьированной траекторий, соответственно. Физический смысл данной метрики заключается в том, что затраты топлива для двух траекторий должны быть примерно одинаковыми, т.к. решается задача поддержания. Возьмем допустимое значение S_m равное 1%. Таким образом метрики S_d и S_m однозначно определяют $\boldsymbol{\varepsilon}_1$ и $\boldsymbol{\varepsilon}_2$.

Определив допустимую окрестность вариации для начального фазового вектора, распределим в ней конечный набор начальных условий следующим образом: компоненты векторов $\boldsymbol{\varepsilon}_1$ и $\boldsymbol{\varepsilon}_2$ будем выбирать из равномерного

распределения с нулевым математическим ожиданием. Для каждого начального условия из полученного набора проинтегрируем с постоянным шагом соответствующую траекторию в рамках невозмущенной модели. Таким образом, для управляющей нейронной сети обучающая выборка будет состоять из фазовых векторов и сопряженных к скорости векторов, задающих управление: $X_{train} = \{X_i\}_{i=1}^M$, $Y_{train} = \{Y_i\}_{i=1}^M$, где $X_i = \{[\mathbf{r}_n^i, \dot{\mathbf{r}}_n^i]\}_{n=1}^N$, $Y_i = \{\lambda_n^i\}_{n=1}^N$, M – количество траекторий, а N – количество точек на траектории. На рисунке 2 представлен набор траекторий, построенных относительно номинальной траектории. На основе данного набора формируется обучающая выборка для задачи управления.

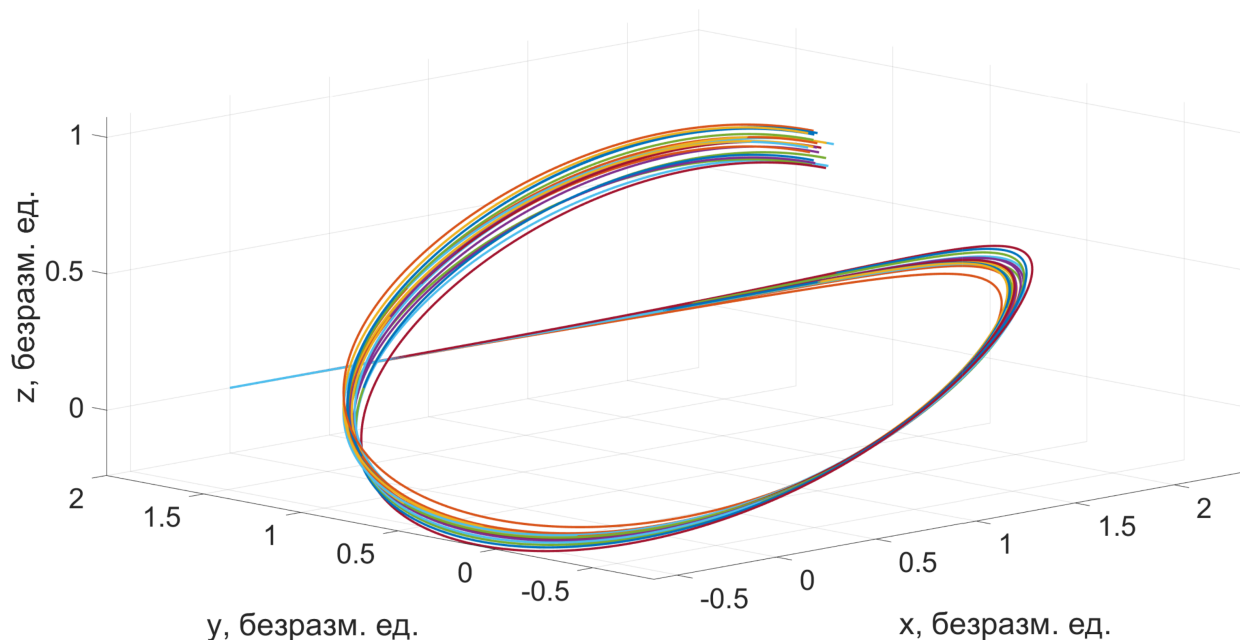


Рисунок 3 – Набор траекторий для обучающей выборки задачи управления.

Перейдем к формированию обучающей выборки для задачи прогнозирования оставшегося времени. Также как и для задачи управления, область фазового пространства, в которой будет составляться выборка, будет ограничена. Уточнение оставшегося времени полета имеет смысл только при подлете к цели, поэтому будем рассматривать только последний виток траектории. Для этого возьмём номинальную траекторию и аналогично построим относительно нее трубку траекторий. Теперь будем брать точки на послед-

нем витке траекторий, соответствующие системе (2), и рассматривать их как начальное условие \mathbf{z}_0 задачи Коши, только с нефиксированным временным концом. При интегрировании уравнений будем смотреть невязку $\|\mathbf{r}(t) - \mathbf{r}_1\|$ и фиксировать T при котором невязка будет минимальна. Таким образом, для нейронной сети, прогнозирующей движение, обучающая выборка будет состоять из фазовых векторов, сопряженных к скорости векторов, задающих управление и уточненного времени полета: $X_{train} = \{X_i\}_{i=1}^M$, $Y_{train} = \{Y_i\}_{i=1}^M$, где $X_i = \{[\mathbf{r}_n^i, \dot{\mathbf{r}}_n^i, \lambda_n^i]\}_{n=1}^N$, $Y_i = \{T_n^i\}_{n=1}^N$, M – количество траекторий, а N – количество точек на траектории.

Составив обучающие выборки, необходимо нормализовать данные. Пусть выборка будет устроена следующим образом: строка будет представлять обучающий пример, а столбец – признак, например, эволюцию одной из компонент радиус-вектора. Будем нормализовать каждый столбец выборки в соответствии с формулой $x_{norm} = \frac{x - \mu}{\sigma}$, где μ – среднее по столбцу, σ – стандартное отклонение по столбцу, x – элемент столбца. Для обратного преобразования необходимо будет запомнить все средние и стандартные отклонения по столбцам. В итоге полученные значения по столбцам будут лежать на отрезке $[-1, 1]$. Данное свойство способствует более точному расчету градиентов во время обучения ИНС. Обратное преобразование состоит в применении очевидной формулы: $x = \sigma x_{norm} + \mu$.

4.3. Архитектура ИНС

После того как обучающие выборки нормированы, 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. В 1957 году А.Н. Колмогоровым и В.И. Арнольдом была доказана теорема о представлении непрерывных функций нескольких пере-

менных суперпозицией непрерывных функций одной переменной, которая в 1989 году была переформулирована и доказана Д. Цыбенко [17] 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. Одной из целей данной работы является исследование минимального размера сетей, т.е. количества слоев и нейронов. Достаточное количество слоев прямо следует из данной теоремы, необходимое же количество слоев и нейронов будет исследовано в следующей главе.

Функционал ошибки выберем из физических соображений. Поскольку обе нейронные сети должны минимизировать евклидово расстояние, то в качестве функционала выберем среднеквадратичное отклонение:

$$E_{MSE} = \frac{1}{2N} \sum_{n=1}^N (\mathbf{y}_n - \mathbf{y}'_n)^2,$$

где N — количество примеров в обучающей выборке, \mathbf{y}_n — выходной обучающий пример, а \mathbf{y}'_n — полученный отклик сети. Теперь определимся с активационными функциями. Для скрытого слоя в качестве нелинейной активационной функции выберем гиперболический тангенс $\phi(x) = \text{th}(x)$. Он является непрерывно дифференцируемым и монотонным, поэтому удовлетворяет определению активационной функции. Полученная нейронная сеть называется сетью прямого распространения.

Рассмотрим выход произвольного скрытого слоя выбранной сети. Пусть в $n - 1$ слое M нейронов, тогда i -я компонента выхода n -го скрытого слоя будет записана следующим образом: $y_i = b_i + \phi(\sum_{j=1}^M \omega_{ij}x_j)$, где $\boldsymbol{\omega} = (\omega_{ij})$ - матрица весов, связывающая $n - 1$ и n -й скрытые слои, \mathbf{x} - выход $n - 1$ скрытого слоя и \mathbf{b} - смещение, вектор — компоненты которого взяты

из равномерного распределения на $[-1, 1]$.

Выход последнего слоя записывается в следующем виде: $\mathbf{y}' = \mathbf{b} + \boldsymbol{\omega}^T \mathbf{x}$, где \mathbf{x} - выход последнего скрытого слоя. Разница между скрытыми слоями и выходным заключается в том, что к последнему слою не применяется функция активации.

В качестве градиентного метода будет выбран стохастический градиентный спуск. В машинном обучении градиент функции потерь обычно рассчитывают, как сумму градиентов, посчитанных на каждом элементе обучения. Поэтому стандартному градиентному спуску требуется один проход по обучающей выборке до того, как он сможет поменять веса. Приведем пример одной итерации изменения весов, связывающих последний скрытый слой и выходной слой, при стандартном градиентном спуске для нашего функционала ошибки:

$$\boldsymbol{\omega}_{i+1} = \boldsymbol{\omega}_i - \alpha \frac{\partial E_{MSE}}{\partial \boldsymbol{\omega}_i}.$$

При стохастическом градиентном спуске значение градиента аппроксимируются градиентом функции ошибки, вычисленном только на одном элементе обучения. Затем веса изменяются пропорционально приближенному градиенту. Таким образом веса модели изменяются после каждого объекта обучения. Пример итерации в случае стохастического градиентного спуска для того же функционала:

$$\boldsymbol{\omega}_{i+1} = \boldsymbol{\omega}_i - \alpha \frac{\partial (\mathbf{y}_n - \mathbf{y}'_n)^2}{\partial \boldsymbol{\omega}_i},$$

где n - номер случайно взятого примера из выборки. Для больших обучающих выборок стохастический градиентный спуск может дать значительное преимущество в скорости по сравнению со стандартным градиентным спуском. Между этими двумя видами градиентного спуска существует ком-

промисс. Он заключается в том, что градиент аппроксимируется суммой градиентов для некоторого количества обучающих примеров.

5. Моделирование

Одной из основных целей данной работы является исследование минимального размера полученных сетей, при котором они способны удовлетворять заданной точности. Допустимую точность попадания в цель положим равной 1 км. Подойдем к решению данной проблемы в два этапа: сначала проверим влияние количества слоев на точность попадания КА в цель в возмущенной модели, далее для полученной сети проверим влияние количества нейронов на той же метрике. Будем тестировать модель на всех основных возмущениях, которые действуют на аппарат: силы сопротивления атмосферы, гармоники 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 $\mu_S = 1.3271244 \times 10^{11} \text{ км}^3/\text{с}^2$ – 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 $\mu_M = 4.9028002 \times 10^3 \text{ км}^3/\text{с}^2$ – 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

$$f = \frac{c_x A v^2}{2M} \rho_l \left(\frac{\rho_h}{\rho_l} \right)^{(D-D_0)/(D_1-D_0)} \quad (3)$$

Здесь $c_x = 2.2$ – коэффициент лобового сопротивления, $A = 60$ м² – характерная площадь поперечного сечения КА. Формула (3) 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 4. В результате воздействия возмущений, параметры орбиты в 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

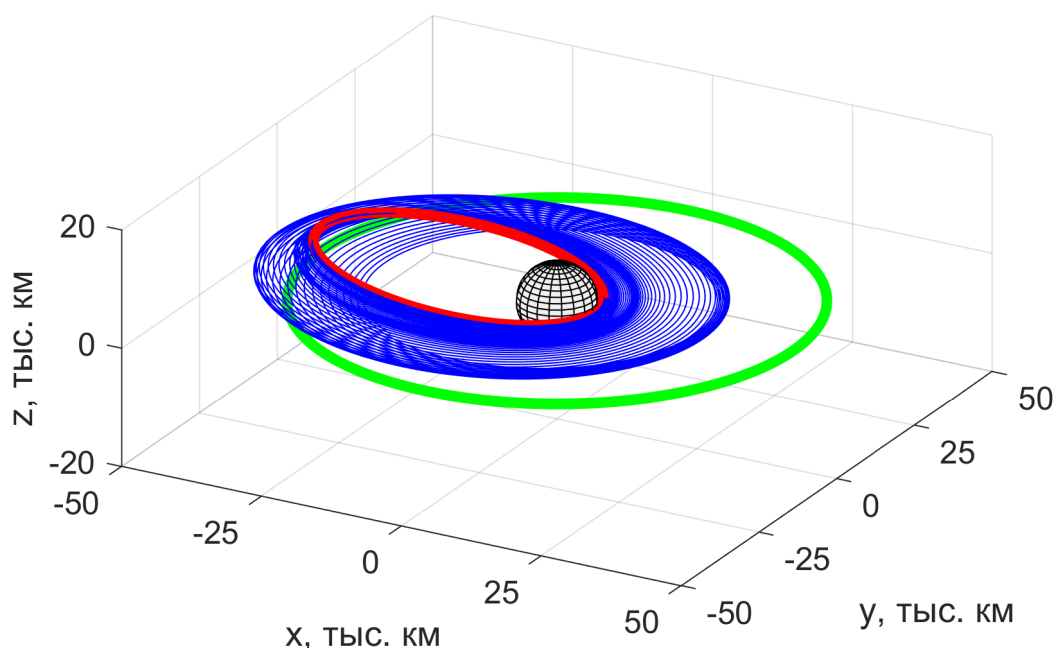


Рисунок 4 – Многовитковая возмущенная номинальная траектория между геопереходной орбитой (красная) и геостационарной орбитой (зеленая); время полета 35 дней, всего совершается 50 витков вокруг Земли (в центре).

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.

Рассмотрим теперь зависимость точности попадания КА в цель от количества слоев в сети. Будем фиксировать количество нейронов в слоях и тестировать на модельных возмущениях. Приведем формулу расчета количества весов для управляющей ИНС: на вход подается вектор $\mathbf{x} \in R^6$, выходной вектор $\mathbf{y} \in R^3$. Пусть в сети n скрытых слоев, в каждом из которых по M нейронов, тогда в сети будет $N = 6 * M + (n - 1) * M^2 + M * 3$ весов. Видно, что количество весов в сети, а следовательно, и количество операций, возрастает экспоненциально от количества слоев, поэтому сперва нужно рассмотреть нейронную сеть, у которой будет только один скрытый

слой. По теореме Цыбенко можно найти такое количество нейронов, при котором ИНС будет способна аппроксимировать искомое отображение. Проведем серию экспериментов для моделей с разным количеством нейронов, начиная от 50, заканчивая 1000, каждый раз увеличивая общее количество нейронов на кратное 50 число. Отметим на рисунке 5 зависимость точности попадания КА в цель от числа нейронов для ИНС с одним скрытым слоем.

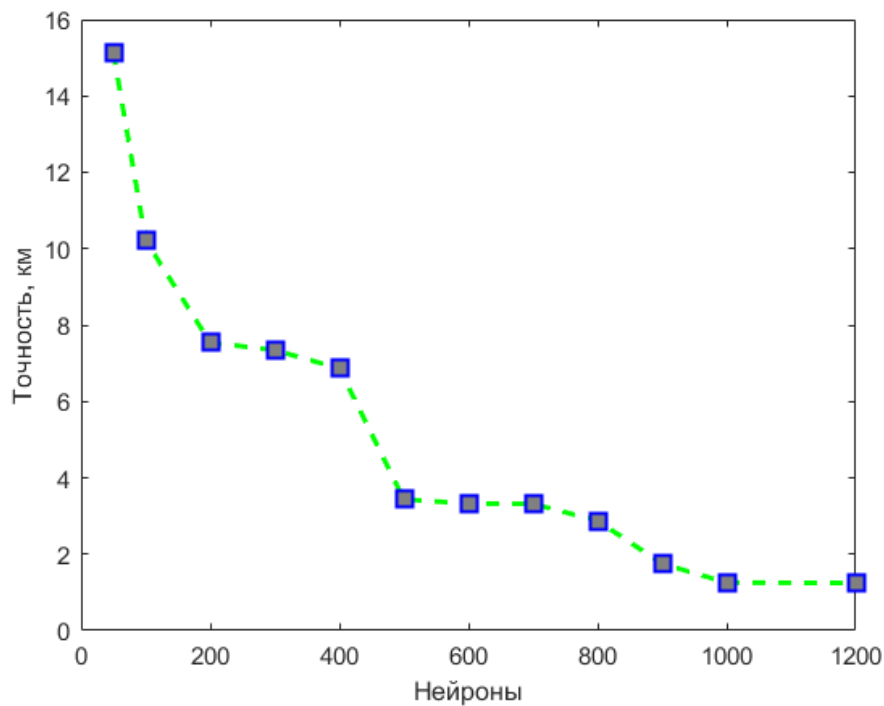


Рисунок 5 – Точность попадания КА в цель от числа нейронов для ИНС с одним скрытым слоем.

После оптимизации числа нейронов был получен следующий результат – модели с одним скрытым слоем начинают удовлетворять заданной точности при достаточно большом количестве нейронов ~ 1000 , однако при таком подходе невязка выходит на асимптоту, совпадающую с максимально допустимой точностью. Для одного скрытого слоя можно зафиксировать полученное число как минимальное. Соответствующее число весов $N = 6 * 1000 + 1000 * 3 = 9000$. Теперь проделаем ту же серию экспериментов, но уже для ИНС с двумя слоями. Для каждого слоя будем брать одинаковое количество нейронов, начиная с 25 и заканчивая 200. Шаг будем

выбирать адаптивно, т.е. при сильном изменении невязки, будем уменьшать добавленное количество нейронов.

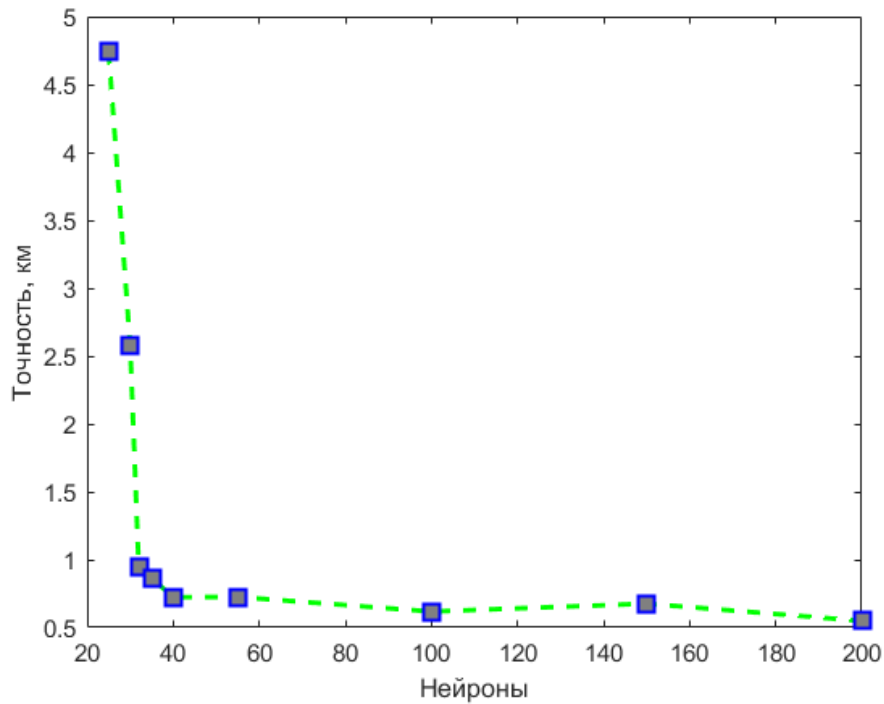


Рисунок 6 – Точность попадания КА в цель от числа нейронов для ИНС с двумя скрытыми слоями.

На рисунке 6 видно резкое уменьшение невязки при достижении $M = 30$, поэтому начиная с данной конфигурации будем увеличивать M на 2. Следующая модель с $M = 32$ удовлетворяет заданной точности, а общее число весов $N = 6 * 32 + 32^2 + 32 * 3 = 1312$. Фиксируем данную конфигурацию, как доставляющую минимум невязке при минимально возможном количестве весов. Полученное количество весов почти в 7 раз меньше, чем соответствующий минимальный показатель для модели с одним скрытым слоем, при этом модель с двумя слоями еще и обеспечивает невязку в среднем в два раза меньше.

Наконец, проделаем все эксперименты для сетей с тремя слоями. Для каждого слоя также будем брать одинаковое количество нейронов, начиная с 15 и заканчивая 50 с адаптивным шагом.

На рисунке 7 видно, что тенденция убывания невязки такая же как и

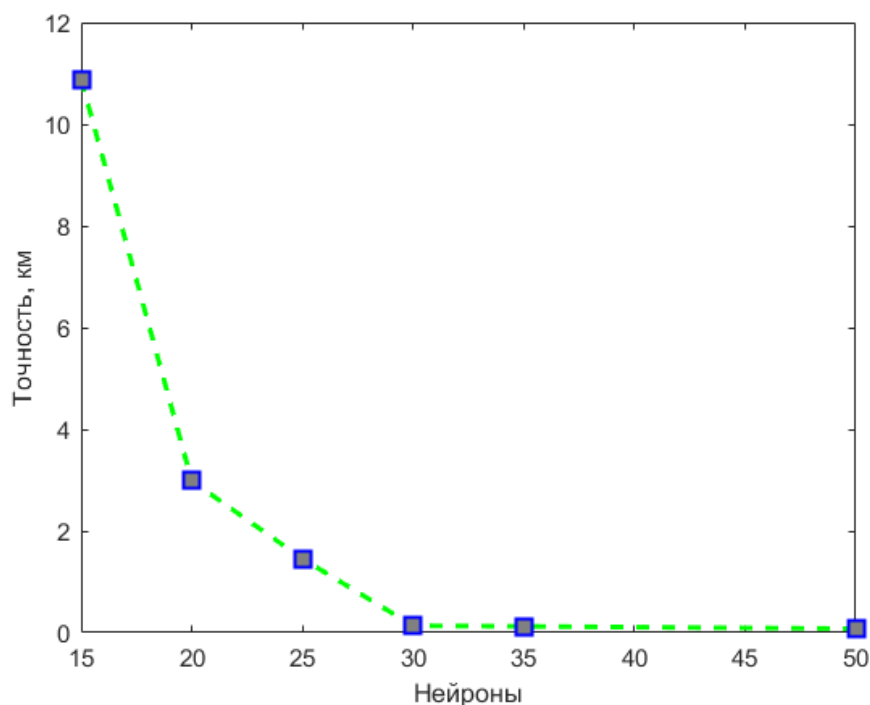


Рисунок 7 – Точность попадания КА в цель от числа нейронов для ИНС с тремя скрытыми слоями.

у моделей с двумя скрытыми слоями, разница заключается в том, что уже для $M = 50$ нейронов невязку получается сократить до 100 метров. Число нейронов M , при котором выполняется условие на невязку равно 30. Однако при такой конфигурации количество весов $N = 6 * 30 + 2 * 30^2 + 30 * 3 = 2070$. Это в полтора раза больше, чем соответствующее значение для сети с двумя скрытыми слоями.

На основе проведенных экспериментов можно сделать вывод, что сеть с 2-мя скрытыми слоями, по 32 нейрона в каждом, является лучшей с точки зрения точности и количества весов.

Найдя минимальную сеть, продолжим моделирование с ее использованием. 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 8.

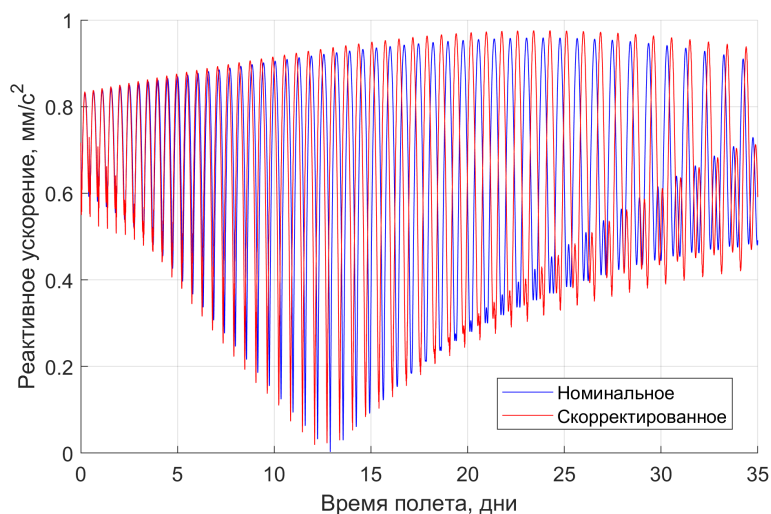


Рисунок 8 – Реактивное ускорение при движении по номинальной траектории в невозмущенной модели (синяя кривая) и реактивное ускорение, выдаваемое нейронной сетью в возмущенной модели.

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 9. Хорошо видно,

итоговые затраты на поддержание составляют порядка 8 кг, что меньше 0.3% от массы аппарата и 1% от затрат топлива на перелет.

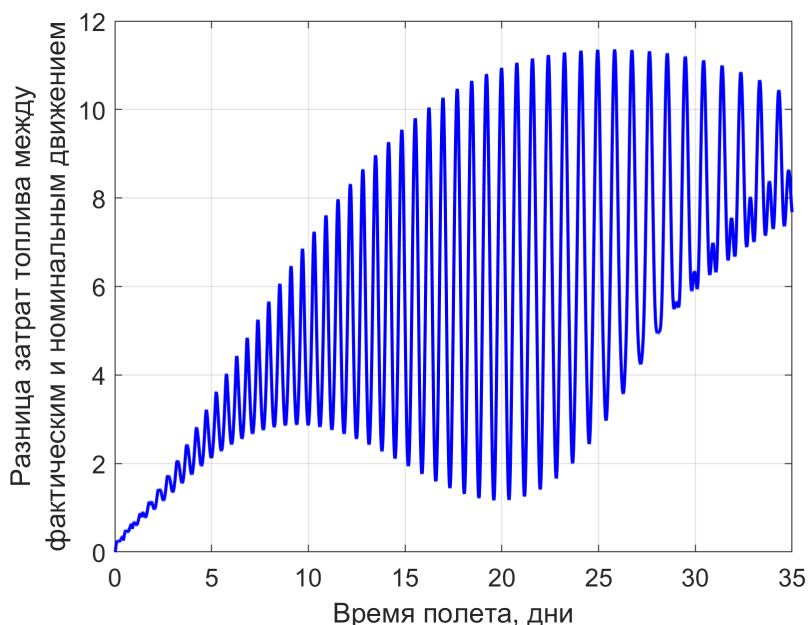


Рисунок 9 – Разница между массой аппарата с нейронной сетью в возмущенной модели и массой аппарата на номинальной траектории в невозмущенной модели в зависимости от времени полета.

Теперь перейдем к проверке модели на отказоустойчивость. Будем понимать этот термин как способность нейронной сети продолжать корректно функционировать в случае отказа каких-то из ее компонент. Исследование нейронных сетей на отказоустойчивость мотивируется тем, что высокоэнергетические частицы в космосе непрерывно бомбардируют КА, бортовой компьютер и способны нарушить работу блока управления. Нарушение работы блока управления мы будем моделировать как выключение произвольного числа случайных нейронов. Также будем менять веса на случайные большие числа.

В качестве тестируемой модели по-прежнему будем брать минимальную сеть, полученную ранее. Сохраним настроенные параметры данной сети и начнем процедуру обнуления весов. Валидацию будем проводить на модельных возмущениях, описанных выше.

Начнем моделирование с обнуления 0.1% общего количества весов.

Постепенно будем увеличивать процент, до тех пор, пока будут соблюдены ограничения на невязку. На рисунке 10 построим зависимость точности попадания КА в цель от процента обнуленных весов. Каждой точке на графике будет соответствовать усредненное значение невязки по числу экспериментов с зафиксированным процентом отключенных весов. Для каждого значения процента будем проводить 10 случайных экспериментов. Также проведем эксперимент и для минимальной трехслойной ИНС отметив результаты на рисунке 10.

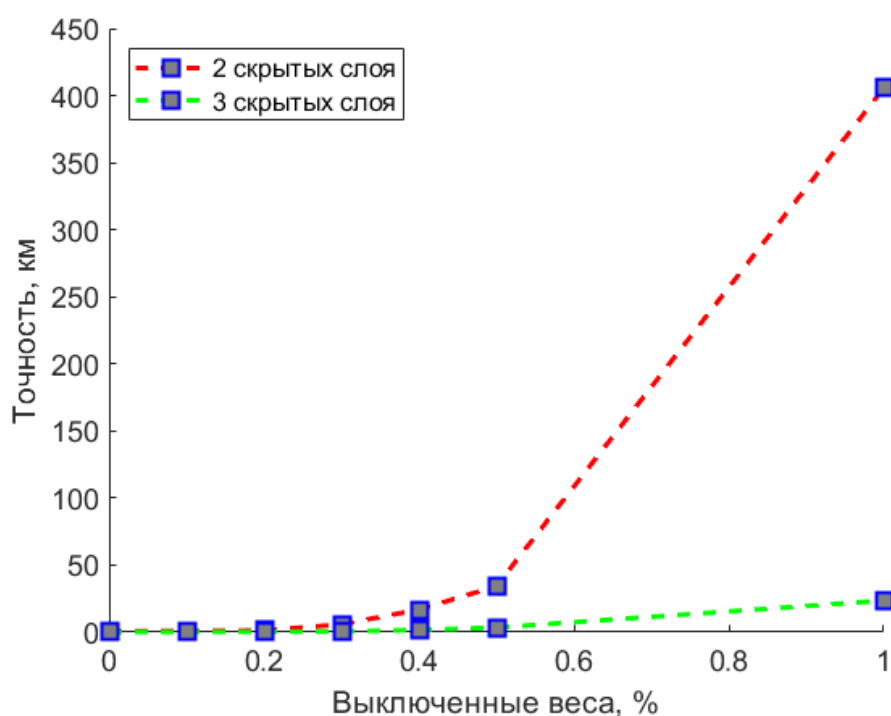


Рисунок 10 – Точность попадания КА в цель от процента выключенных весов для минимальной ИНС.

Видно, что для двухслойной сети уже при 0.2% выключенных весов ограничения не соблюдены и промах больше километра, а это всего 3 веса в сети. Значения невязки для трехслойной сети растут на порядок медленнее. Таким образом можно сделать вывод, что свойство отказоустойчивости ИНС проявляется при увеличении количества слоев.

Заключение

Разработаны две искусственные нейронные сети (управляющая и прогнозирующая), которые могут быть использованы для управления орбитальным движением КА вблизи спиралевидной номинальной траектории. Для управляющей сети выборка формировалась в результате решения совокупности краевых задач принципа максимума Понтрягина методом стохастического градиентного спуска в окрестности номинального решения. Для прогнозирующей сети выборка формируется в результате решения множества задач Коши. Обучение нейронных сетей проводилось в рамках невозмущенной задачи двух тел. Тестирование нейронных сетей в возмущенной задаче двух тел показало: 1) наличие нескольких скрытых слоев дает лучший результат аппроксимации искомого отображения, нежели единственный слой, 2) удовлетворять ограничению на невязку может относительно небольшая нейронная сеть состоящая из двух слоев, в каждом слое которой 32 нейрона, 3) свойство отказоустойчивости можно наблюдать в сетях с большим количеством слоев. Кроме того, рассмотрена модельная задача перелета между геопереходной и геостационарной орбитами в рамках возмущенной задачи двух тел, учитывающей возмущения от сил атмосферного сопротивления, гармоника 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.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [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] Шишкин О. Г. Искусственная нейронная сеть для диагностики космического аппарата // Интеллектуальные системы, управление и мехатроника-2016. 2016. С. 448–451.
- [12] 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.
- [13] 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.
- [14] 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.
- [15] Петухов В.Г. Оптимизация межпланетных траекторий космических аппаратов с идеально-регулируемым двигателем методом продолжения // Космические исследования. 2008. Т. 46, № 3. С. 224–237.
- [16] Maksim Shirobokov Sergey Trofimov, Ovchinnikov Mikhail. Pareto-Optimal Low-Thrust Lunar Transfers with Resonant Encounters // 3rd IAA

Conference on Dynamics and Control of Space Systems (DyCoSS) Moscow.
30 May – 01 June, 2017. T. 2, № 4. C. 303—314.

- [17] Cybenko G. V. Approximation by Superpositions of a Sigmoidal function // Mathematics of Control Signals and Systems. 1989. T. 2, № 4. C. 303—314.