

АППРОКСИМАЦИЯ ГРАНИЦЫ МНОЖЕСТВА ДОСТИЖИМОСТИ МЕТОДАМИ МАШИННОГО ОБУЧЕНИЯ

А.А. Петров (*aleksand.a.petrov@yandex.ru*)^А

В.В. Латонов (*WLatonov@gmail.com*)^В

^А Московский государственный университет им. М.В. Ломоносова,
Москва

^В ПАО «Сбербанк»

В данной работе решается задача оценки множества достижимости динамической системы при помощи методов машинного обучения и нейросетей. Этими средствами возможно решить такую задачу быстрее, чем классическими численными методами, такими, как пиксельный метод. В работе приведены два метода аппроксимации границы множества достижимости: метод опорных векторов (SVM) и полносвязная нейросеть с одним скрытым слоем, которые сравниваются с пиксельным методом. Приведена математическая модель «машина Дубинса» в различных постановках, которая используется в качестве примера поиска множества достижимости. Описаны метод генерации данных с помощью метода Чебышева и схемы формирования обучающих выборок. На основе численных экспериментов проведен анализ точности и быстродействия методов, их поведения в окрестности границы и чувствительности к способу формирования выборки. Итоговый рассмотренный нейросетевой метод обеспечил точность в 93% относительно пиксельного метода, при этом увеличив скорость работы более, чем в 3000 раз, что говорит о применимости нейросетевых моделей для оценки множества достижимости в реальном времени (например, для спутников или автономных систем).

Ключевые слова: множество достижимости; машина Дубинса; аппроксимация границы; пиксельный метод; метод опорных векторов; нейросети; дифференциальные нейросети.

Введение

Задача оценки множества достижимости (reachability) динамической системы заключается в определении всех состояний, которые могут быть достигнуты из заданного начального состояния при всех допустимых управлениях за ограниченный ресурс (например, за заданное время или при ограничении на длину пути). Множество достижимых состояний (часто рассматривается его граница) является важнейшим понятием в теории управления, робототехнике и анализе безопасности, поскольку позволяет ответить на вопросы о том, какие области пространства состоятельны для движения системы, а какие *не* могут быть достигнуты (например, из-за ограничений на управление). Оценка множества достижимости находит применение в синтезе безопасных траекторий движения автономных устройств, в проверке корректности работы систем управления, при анализе столкновений и в реальном времени может выступать инструментом для принятия решений об ограничениях управления.

Современные подходы к задаче достижимости (reachability) все чаще делают акцент на *аппроксимации* множества достижимости, жертвуя строгой точностью ради существенного выигрыша в скорости вычисления. Одним из перспективных направлений является использование методов *машинного обучения* и классификации. Идея заключается в том, чтобы рассматривать задачу принадлежности точки пространству достижимых состояний как задачу бинарной классификации, которую можно решить с помощью модели, обученной на заранее сгенерированных данных. Однако в большинстве случаев всё ещё используются классические методы решения, несмотря на запрос по увеличению скорости обработки, так как методы, подобные описанному выше, не дают достаточных гарантий, что критично в большинстве задач управления. Таким образом, остаётся потребность в разработке подходов, которые могли бы совместить преимущества машинного обучения и гарантировать достаточную надежность и стабильность решений, что и определяет актуальность данной работы. В представленной работе решена задача быстрого поиска множества достижимости, которая была освещена в упрощённом виде в работе [Лотов, 2018], где быстрый подсчёт множества достижимости был достигнут за счёт геометрических свойств системы, что бывает крайне редко.

Обзор литературы

Формальное введение множества достижимости и основные подходы к его определению рассмотрены, например, в [Maler, 2008]. Классические методы вычисления таких множеств включают уравнения Гамильтона–Якоби и перебор траекторий, но они требуют значительных ресурсов. Поэтому разрабатываются приближённые алгоритмы. В [Лотов, 1998] пред-

ложен дискретный подход (метод достижимых целей), в котором граница множества аппроксимируется на сетке с последовательным уточнением. В [Горнов, 2015] предложен пиксельный метод с кусочно-линейной аппроксимацией границы, подходящий для систем средней размерности, но теряющий эффективность при росте размерности из-за увеличения числа граней.

Современные исследования направлены на ускорение вычислений в задачах реального времени. В [Allen, 2017] применяются методы машинного обучения для быстрого приближения достижимых множеств, что полезно для онлайн-планирования. В [Bui, 2021] разработана FPGA-реализация алгоритма решения уравнений Гамильтона–Якоби, позволяющая вычислять 4D-множества с частотой до 4 Гц. В [Holmes, 2020] метод ARMTD использует заранее рассчитанные множества для безопасной онлайн-оптимизации траекторий манипулятора. Для низкоразгонных космических аппаратов в [Bowerfind, 2024] предложен семплинг-алгоритм с использованием оптимального управления, который ускоряет расчёт множеств и хорошо согласуется с динамикой системы. В [Shao, 2022] рассмотрена задача относительного движения спутника; предлагается полиэдральное приближение множества достижимости с учётом ограничений управления.

Нейросети активно применяются для компенсации неопределённостей и наблюдения. В [Cheng, 2021] описан адаптивный контроллер, использующий онлайн-оценку нейросетью для компенсации нелинейностей, обеспечивая точность и устойчивость. В [Alfaro-Ponce, 2018] предложен наблюдатель на основе дифференциальной нейросети, устойчивый к квантованию выходных данных. В [Hasan, 2023] нейросеть интегрирована в смешанный наблюдатель для оценки состояния автомобиля с экспериментальным улучшением точности. Работа [Xiang, 2018] посвящена верификации нейросетей: предложен метод вычисления выходного диапазона многослойного перцептрона, критичный для безопасного применения ИИ в управлении.

Методы аппроксимации границы множества достижимости

После того, как будут подготовлены тренировочные данные (координаты точек и метки их достижимости), можно применять различные методы классификации для аппроксимации границы множества достижимых состояний. В данной работе рассматриваются три подхода.

Пиксельный метод. Пиксельным методом называется алгоритм прямого вычисления множества достижимости на дискретной сетке состояний. Суть метода заключается в следующем: пространство (x, y, v) (или соответствующие подпространства, например, (x, y)) квантуется с заданным шагом по каждой координате, образуя сетку узлов (пикселей в конечномерном пространстве). Затем запускается итеративный процесс

распространения “волны” достижимых состояний от начального состояния. Изначально помечается достижимым начальный узел, соответствующий (x_0, y_0) . Далее на каждом шаге времени t (дискретизация времени) для всех узлов, помеченных как достижимые на предыдущем шаге, вычисляются их возможные переходы за Δt при всех дискретных вариантах управления (например, для угловой скорости можно перебрать значения ω для скорости v – значения 0 или $\pm v_{max}$, и т.п.). Конечные узлы этих переходов (после интервала Δt) помечаются как достижимые на новом шаге. Процесс повторяется, пока не исчерпан весь горизонт (то есть выполнено N итераций). В результате получается множество узлов сетки, помеченных достижимыми.

Для повышения точности шаги Δt и пространственный шаг сетки берутся достаточно мелкими. Вычислительная сложность пиксельного метода зависит от размерности пространства и требуемой точности. Она пропорциональна числу рассматриваемых узлов, числу шагов времени и числу дискретных управлений. Если обозначить через N количество квантований по каждой переменной, то число пространственных узлов N^d (для моделей 1–2 можно считать $d=1$). Число временных шагов N_t . Число управляющих воздействий N_u – конечное и даёт сложность порядка $N^d N_t N_u$. Даже для умеренных значений

операций, что крайне много. Его главное достоинство – концептуальная простота и наглядность: по сути, он моделирует “заливку” области. В наших экспериментах результаты пиксельного метода (при очень мелкой сетке) рассматриваются как эталонное множество достижимости, с которым сравниваются остальные методы. Однако для применения в реальном времени такой метод не подходит из-за больших вычислительных затрат.

Методы машинного обучения. В работе применялись два подхода машинного обучения для классификации точек на достижимые и недостижимые: метод опорных векторов (SVM) и двухслойная полносвязная нейронная сеть (MLP).

При использовании SVM точки обучающей выборки классифицировались при помощи радиально-базисного ядра (RBF), выбранного из-за явно нелинейной и изогнутой границы множества достижимости. Обучение SVM заключалось в решении задачи квадратичной оптимизации, в результате которой получались опорные векторы, определяющие разделяющую поверхность.

Для нейросетевого подхода использовалась архитектура MLP с одним скрытым слоем, количество нейронов которого варьировалось от 5 до 20 и подбиралось по результатам на валидационной выборке. Входом сети являлись координаты точек (x, y) , выходом – один нейрон с сигмоидной

активацией, выдающий вероятность принадлежности точки к достижимому классу. Скрытый слой использовал нелинейные функции активации (сигмоидальную, \tanh , а также ReLU). Обучение сети проводилось методом градиентного спуска с оптимизатором Adam на тех же размеченных данных, что и для SVM.

Математическая модель

Модель «машина Дубинса». В качестве тестовой динамической системы выбрана модель «машина Дубинса» [Munts, 2020] – движение в горизонтальной плоскости с ограничением на скорость поворота и скорость движения, описываемая данной системой:

где управление включает угловую скорость ω и управление ускорением a . Для модели задается начальное состояние системы. В данной работе предполагается фиксированное начальное положение и начальный курс θ_0 (ориентация вдоль оси Ox). Множество достижимости определяется как совокупность всех состояний (x, y, θ) которых система может достичь хотя бы при одной допустимой реализации управления за заданный промежуток времени $[0, T]$. В частности, основной интерес представляет проекция этого множества на плоскость Oxy – зона достижимости в геометрическом пространстве. В дальнейших рассуждениях под *множеством достижимости* будем понимать именно проекцию на плоскость (x, y) за фиксированное время T , поскольку ориентация и скорость конечного состояния не столь важны для геометрической зоны достижимости, хотя необходимо учитывать, что фактическая размерность задачи имеет значение 4 – две координаты, угол движения и модуль скорости.

Ограничения и параметры. В типичных сценариях задаются численные ограничения, например: максимальная угловая скорость ω_{max} (что эквивалентно минимальному радиусу поворота $R_{min} = V/\omega_{max}$ для варианта 2), максимальная скорость V_{max} (для варианта 3), а также горизонт T (время, за которое рассматривается достижимость). В нашем исследовании можно считать, что $\omega_{max} = 1$ (единичная скорость в безразмерных единицах), $R_{min} = 1$ (единица угловой скорости, что дает минимальный радиус поворота $R_{min} = 1$ для варианта 2), а горизонт T – некоторый фиксированный промежуток (например, $T = 1$). Все величины можно считать безразмерными после соответствующей нормализации.

Генерация данных. Для применения методов машинного обучения необходимо сформировать выборку примеров состояний, принадлежащих или не принадлежащих множеству достижимости. Проще говоря, требуется множество точек , снабженных меткой класса: 1 (достижимо) или 0 (недостижимо) из начального состояния за время . Поскольку аналитического описания границы множества достижимости для машины Дубинса (в общем случае) не существует, в работе получены *эталонные данные* численным путем. В качестве эталонного способа определения достижимости реализован пиксельный метод.

Схема построения выборок. Получив эталонное множество достижимости (например, в виде списка достижимых узлов сетки или полигона, охватывающего эту область), формируем обучающие выборки для последующей классификации. Рассматриваются два типа выборок:

1. *Равномерная выборка:* точки генерируются равномерно по площади в некоторой области, гарантированно содержащей множество достижимости (например, в круге радиуса вокруг начального положения). Каждой точке присваивается метка 1, если она попала в достижимую область (по данным эталонного пиксельного метода), и 0 – если не попала. Таким образом, формируется сбалансированный набор примеров внутри и вне достижимой зоны.

2. *Выборка с увеличенной плотностью у границы:* в данной работе применялся алгоритм на основе поиска ближайшего соседа – если для точки в пределах радиуса есть точка противоположного класса, то она считается близкой к границе, в противном случае – далёкой от нее. Далее на такой выборке вместо равномерного распределения бралось распределение, где у всех точек, близких к границе, вероятность попасть в итоговую выборку выше, чем у далёких от границы, в раз. В итоге получается выборка, где значительно больше точек расположено вблизи разделяющей поверхности «достижимо/недостижимо». Цель такого подхода – дать обучающему алгоритму больше информации о форме границы, поскольку именно там решается задача классификации (точки далеко внутри или вне области тривиальны для классификатора).

Последовательность шагов, необходимых для генерации и применения выборки, визуально представлена на рис. 1.



Рис. 1. Визуальная блок-схема составления выборки и применения методов

Численные эксперименты

И метод опорных векторов, и нейросеть обучается на данных, полученных пиксельным методом. То есть, они не вычисляют множество достижимости *с нуля*, а именно аппроксимируют уже известное (приближенно) множество. Тем не менее, даже с учетом затрат на предварительную генерацию данных, выигрыш оказывается существенным: данные можно сгенерировать один раз офлайн (например, на этапе проектирования системы), а в процессе работы эксплуатации использовать легковесную модель для классификации новых ситуаций или применять пиксельный метод локального для дообучения моделей.

Для сравнения описанных методов проведены численные эксперименты на модели «машина Дубинса». Эталонное множество достижимости за время получено пиксельным методом на сетке с шагом 0.01 по α и 1° по углу β .

Для обучения моделей SVM и нейросети сгенерировано $\sim 10,000$ точек в квадрате $[-1.2, 1.2]$. Примерно половина точек выбрана равномерно, половина — с упором к границе. Из них 60% использовано для обучения, остальные для контроля и теста (по 20%). SVM настроен с RBF-ядром, параметр ширины ядра и регуляризации подобраны по кросс-валидации на контроле. Нейросеть взята с 10 нейронами скрытого слоя, сигмоидной активацией, обучена на 500 эпох (до сходимости ошибки).

Эти гиперпараметры моделей подбирались методом перебора по сетке (grid search) на валидационной выборке. Реализация моделей выполнена с использованием классической библиотеки Python: Scikit-learn (`sklearn.svm.SVC` с RBF-ядром и `sklearn.neural_network.MLPClassifier`).

но). Архитектура нейросети выбрана максимально простой (один скрытый слой с 10 нейронами) – это минимально достаточная конфигурация, которая, как показали эксперименты, обеспечивает требуемое качество решения задачи на базовом уровне.

Таблица 1

Сравнение точности классификации и времени вычисления
для разных методов

Т	Выборка	Метод	Точность	Время
15с	Равномерная	SVM	67%	0.001с
15с	Смещённая ($R=1$, $\text{prob}=x10$)	SVM	72%	0.001с
15с	Равномерная	NN	93%	0.017с
15с	Смещённая ($R=1$, $\text{prob}=x10$)	NN	91%	0.01с
15с	Равномерная	Пиксельный	*100%	29.8с
15с	Смещённая ($R=1$, $\text{prob}=x10$)	Пиксельный	*100%	35.1с

Для оценки качества классификации использовалась метрика ассигасу, понимаемая как доля правильно классифицированных точек тестовой выборки. В случае нейросетевой модели для равномерной выборки эта доля составила около 93%. Соответствующая матрица ошибок показывает, что ~30% всех тестовых точек были правильно определены как достижимые (True Positive), ~60% – правильно как недостижимые (True Negative), около 4% оказались ложно недостижимыми (False Negative) и ~3% – ложно достижимыми (False Positive). Эти ошибки распределены относительно равномерно и приходятся в основном на точки, лежащие вблизи истинной границы множества достижимости. Подобное явление типично для задач классификации данного типа, поскольку в окрестности границы даже незначительные отклонения условий могут изменить метку достижимости. Для более явного анализа и интерпретации граничных элементов применимы методы вербального анализа решений; в системе ДИФКЛАСС [Ларичев, 1996] задаётся процедура построения непротиворечивых правил дифференциальной классификации, удобная для проверки устойчивости разметки у границы.

В табл. 1 представлены сводные показатели для трех методов: пиксельного (эталон), SVM и нейросети. Точность классификации оценивалась на независимых тестовых данных и характеризует, какой процент точек правильно определен как достижимый/недостижимый (по сравнению с эталоном). Пиксельный метод потребовал порядка нескольких минут вычислений для построения границы. В то же время, проверка на принадлежность множеству достижимости методом SVM заняла ~ с (1 мс) на CPU, а нейросетью – ~ с (10 мс). Иными словами, SVM позволяет получать ответ практически мгновенно, нейросеть – очень быстро

в масштабе реального времени, а вот пиксельный метод непригоден для онлайн-применения (его можно использовать только офлайн или в ограниченном объёме данных). На рис. 1 и 2 представлены примеры разметок тестовой выборки для рассмотренных случаев.

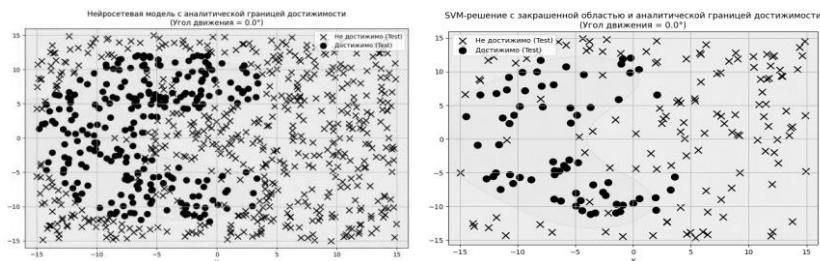


Рис. 2. Аппроксимация границы множества достижимости.
Метод формирования выборки равномерный

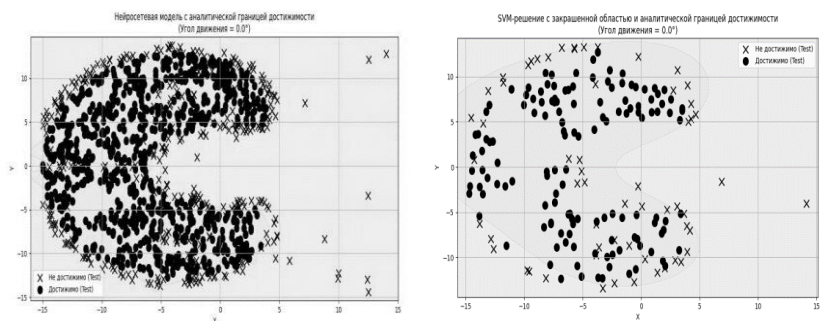


Рис. 3. Аппроксимация границы множества достижимости.
Метод формирования выборки вблизи границы

Выводы

Полученные результаты показывают, что методы машинного обучения способны успешно решать задачу аппроксимации множества достижимости. Основная часть ошибок обеих моделей сконцентрирована вблизи границы области – что естественно, поскольку именно там данные наиболее «сложные»: небольшая погрешность может изменить метку точки. Внутри области и далеко снаружи и SVM, и MLP почти безошибочны (что подтверждается близкой к 100% полнотой/точностью в этих зонах).

В дополнение к базовому сценарию были проведены эксперименты и с другими значениями параметров для нейросетевой модели – в частности, с различными горизонтами времени и максимальными углами поворота

. В табл. 2 приведены показатели качества модели при нескольких таких комбинациях параметров. Видно, что точность классификации остается практически неизменной при варьировании α и γ . Однако обнаружено, что модель, обученная при одних параметрах, показывает существенно низкую точность, если применять её напрямую к данным, сгенерированным при других условиях. В то же время при дообучении предварительно обученной сети под новые параметры для достижения аналогичного качества требуется примерно вдвое меньше итераций обучения, чем при обучении «с нуля». Этот результат указывает на потенциал методов трансферного дообучения и обучения с подкреплением для эффективной адаптации модели к изменяющимся условиям.

Таблица 2

Результаты модели на различных горизонтах прогноза T и ограничениях максимального угла поворота

Горизонт T , сек.	Макс. угол поворота, °	Точность, %
5	30	87
15	30	93
15	60	91
30	60	90

Заключение

Результаты исследования показывают, что методы машинного обучения, в частности SVM и относительно простая нейросетевая модель, эффективно аппроксимируют границу множества достижимости для динамической системы с дифференциальными ограничениями. Нейросетевые модели оказались применимы для оценки достижимости с высокой скоростью и достаточной точностью. Таким образом, новизна данного исследования заключается в том, что, несмотря на потенциальную непредсказуемость нейросетевых моделей, они продемонстрировали одновременно высокое быстродействие и приемлемую точность в задаче аппроксимации границы множества достижимости.

Важным преимуществом подхода является возможность применения в реальном времени за счёт быстродействия разработанного классификатора. Например, автономный дрон или спутник могут принимать решения о допустимых маневрах или реагировать на динамические ограничения при помощи этого классификатора.

Однако следует учитывать и ограничения предложенного подхода. Модель (SVM или нейросеть) обучается на определенных условиях (динамика, ограничения, горизонт времени). Если условия изменятся (на-

пример, изменится ресурс времени или динамические параметры), потребуется либо повторная генерация данных и обучение, либо использование более универсальной модели.

Перспективные направления дальнейших исследований включают адаптацию нейросетей к разным условиям и режимам работы, включающим большее число параметров. Также актуальна переносимость моделей, которую можно обеспечить с помощью трансферного обучения. Ещё одно перспективное направление – интеграция дифференциальных нейросетей, описывающих динамику систем напрямую. Вместо последовательных этапов генерации траекторий и классификации достижимых состояний возможна разработка единой модели, воспроизводящей эволюцию множества достижимости.

Список литературы

- Латонов В.В.** Программные стратегии тестирования качества управления линией визирования по видеонаблюдению // Вестник Московского университета. Серия 1: Математика. Механика. – 2018. – № 6. – С. 51-56.
- Maler O.** Computing Reachable Sets: An Introduction // Hybrid Systems: Computation and Control, LNCS 4981, 2008. – P. 65-76.
- Лотов А.В., Бушенков В.А., Каменев Г.К., Черных О.Л.** Компьютер и поиск компромисса. Метод достижимых целей. – М.: Наука, 1997. – 239 с.
- Горнов А.Ю., Финкельштейн Е.А.** Алгоритм кусочно-линейной аппроксимации границы множества достижимости управляемой системы // Автоматика и телемеханика. – 2015. – № 3. – С. 22-35.
- Allen R.E., Clark A.A., Starek J.A., Pavone M.A.** Machine Learning Approach for Real-Time Reachability Analysis // Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2017.
- Bui M., Lu M., Hojabr R., Chen M., Shriraman A.** Real-Time Hamilton-Jacobi Reachability Analysis of an Autonomous System with an FPGA // Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2021.
- Holmes P., Kousik S., Zhang B., Raz D., Barbalata C., Johnson-Roberson M., Vasudevan R.** Reachable Sets for Safe, Real-Time Manipulator Trajectory Design // Proceedings of Robotics: Science and Systems (RSS), 2020.
- Bowerfind S., Taheri E.** Rapid Approximation of Low-Thrust Spacecraft Reachable Sets within Complex Two-Body and Cislunar Dynamics // Aerospace. – 2024. – 11(5): 380.
- Shao L., Miao H., Hu R., Liu H.** Reachable set estimation for spacecraft relative motion based on bang-bang principle // Chinese Journal of Aeronautics. – 2022. – 36(2). – P. 229-240.
- Cheng L., Wang Z., Jiang F., Li J.** Adaptive neural network control of nonlinear systems with unknown dynamics // Advances in Space Research. – 2021. – 67(3).
- Alfaro-Ponce M., Argüelles A., Chairez I., Avelar A.** Differential Neural Networks Observer for Second Order Systems with Sampled and Quantized Output // IFAC-PapersOnLine. – 2018. – 51(13). – P. 490-495.

- Hasan A.-G., Laghmara H., Ali S. A., Ainouz S., Xing G.** Continuous–Discrete Time Neural Network Observer for Nonlinear Dynamic Systems (Application to Vehicle Systems). – IFAC-PapersOnLine, 2023. – 56(2).
- Xiang W., Tran H.-D., Johnson T.T.** Output Reachable Set Estimation and Verification for Multi-Layer Neural Networks // IEEE Transactions on Neural Networks and Learning Systems. – 2018. – 29(11).
- Munts N.V.** Numerical Study of Different Variants of Dubins' Car Model // Proceedings of the 60th Israel Annual Conference on Aerospace Sciences. – 2020.
- Ларичев О.И., Болотов А.А.** Система ДИФКЛАСС: построение полных и непротиворечивых баз экспертных знаний в задачах дифференциальной классификации // Науч.-техн. информация. Сер. 2: Информационные процессы и системы. – 1996. – № 9. – С. 9-15.