

Машинное обучение систем интеллектуального управления

А. И. Дивеев¹

Рассматривается машинное обучение интеллектуальных систем управления методами символьной регрессии. Методы символьной регрессии позволяют находить математические выражения для различных задач, где необходимо найти структуру и параметры неизвестной многомерной функции. Поиск неизвестной функции осуществляется генетическим алгоритмом на пространстве кодов метода символьной регрессии. В качестве искомым функций могут быть функции, содержащие операторы условия, которые являются обязательной составной частью программ интеллектуальных систем управления.

Ключевые слова: методы символьной регрессии, синтез управления, машинное обучение, оптимальное управление.

1. Введение

Сегодня искусственный интеллект, несмотря на большое количество определений, является программой, работающей в вычислительном устройстве. Основным элементом программы искусственного интеллекта является оператор условия "if". Следовательно, если мы говорим о машинном обучении искусственного интеллекта, то это означает использование математических конструкций, которые в автоматическом режиме создают операторы условия. Формально оператор условия можно получить из ступенчатой функции Хэвисайда или функции знака. Например, программный оператор $\text{if } a < 0 \text{ then } y \leftarrow b \text{ else } y \leftarrow c$, записывается с помощью функции Хэвисайда $\vartheta(a)$ следующим образом $y = (1 - \vartheta(a))b + \vartheta(a)c$. Но следует отметить, что и математических конструкций, в которых в качестве решений, например при интегрировании каких-то дифференциальных уравнений, появляются функции Хэвисайда, также нет.

Допустим рассматривается задача автоматической парковки автомобиля при условии, что на стоянке находятся другие автомобили. Иногда возможен плавный маневр без переключения направления движения.

¹ Дивеев Асхат Ибрагимович — главный научный сотрудник Федерального исследовательского центра "Информатика и управление" Российской Академии наук, e-mail: aidiveev@mail.ru.

Diveev Askhat Ibragimovich — chief researcher, Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences

Но в большинстве случаев плотная парковка требует движения вперед-назад, при этом количество переключений зависит от расположения автомобилей, их габаритов и других факторов. Заранее не известно, сколько операторов условия должно быть в решении. Чтобы найти решение необходимо найти функцию управления, которая из разных начальных состояний переводит объект в терминальное состояние с учетом фазовых ограничений. Формально рассматривается задача общего синтеза управления [1]. Проблема ее решения состоит в том, что функция управления является разрывной с заранее неизвестным количеством точек разрыва.

Ранее при поиске функции исследователь на основе анализа и интуиции задавал ее структуру с точностью до значений параметров. Далее с помощью вычислительных методов находил оптимальные значения этих параметров. Возможно использование универсальных параметрических аппроксимирующих функций, структура которых регулярно усложняется в зависимости от сложности задачи. Для одномерных функций – это ряды, число членов которых зависит от сложности задачи. Для аппроксимации многомерных функций в последнее время используются искусственные нейронные сети, в которых при обучении ищется большое количество параметров, а число слоев нейронной сети зависит от сложности аппроксимируемой функции. Усложнение задач интеллектуального управления приводит к существенному увеличению количества искомых параметров и делает невозможным параметрический поиск функции или параметрическое обучение.

Методы символьной регрессии, появившиеся в конце двадцатого века [2], для автоматического написания программ позволяют искать структуру и параметры функции одновременно. Все методы символьной регрессии кодируют математическое выражение в форме специального кода и ищут решение генетическим алгоритмом на пространстве кодов. Методы символьной регрессии использовались для решения задач управления для синтеза блока управления на основе типовых блоков [3], но не для поиска математического выражения функции управления. В работе [4] предложено искать функцию управления методом генетического программирования и этот процесс назван машинным обучением управления, хотя в работе не представлено ни одной решенной задачи синтеза управления.

В настоящее время существует более десяти методов символьной регрессии [5], [6], которые успешно применяются для решения задачи общего синтеза управления или в современном тренде машинного обучения управления.

2. Методы символьной регрессии

Первый метод символьной регрессии, метод генетического программирования, кодирует математическое выражение в виде вычислительного дерева. Для кодирования необходимо иметь базовое множество элементарных функций, которые размещаются в узлах дерева, а на ветвях дерева размещаются аргументы и параметры. Генетический алгоритм при выполнении операции скрещивания отбирает два возможных решения, устанавливает случайно узлы для скрещивания и обменивает поддерева, начинающиеся с этих узлов. В результате получаем два новых дерева, которые являются кодами новых математических выражений. Генетическое программирование имеет существенный недостаток, коды разных математических выражений имеют разную длину.

Для устранения недостатков генетического программирования были созданы другие методы символьной регрессии. Метод Декартового генетического программирования [7] кодирует математическое выражение в виде наборов целочисленных векторов, первая компонента которого указывает на номер элементарной функции, а остальные компоненты на номера аргументов. Если аргументов больше, чем требуется для функции, то остальные аргументы не используются. После вычисления элементарной функции ее результат добавляется ко множеству аргументов и может быть использован в последующих вызовах функций. Коды Декартового генетического программирования имеют одинаковую длину, при этом часть кода не используется.

Основной недостаток большинства методов символьной регрессии - нарушение свойства наследования при выполнении операции скрещивания. Новые возможные решения, полученные после выполнения операции скрещивания, не оказываются непохожими на своих "родителей". Это обстоятельство соответствует генерации новых возможных решений и эволюционный поиск становится случайным.

В 2006 году был разработан метод сетевого оператора [8] специально для решения задачи синтеза управления. В методе использовался принцип малых вариаций базисного решения. Согласно этому принципу кодом сетевого оператора кодируется только одно базисное решение. Остальные возможные решения кодируются как малые вариации базисного решения. Каждая малая вариация кодируется целочисленным вектором. Каждое возможное решение представляет собой набор целочисленных векторов малых вариаций. Операция скрещивания выполняется на наборах векторов малых вариаций.

Принцип малых вариаций базисного решения является универсальным [9], [10] и может применяться для решения различных нечисловых

задач оптимизации, в которых метрика пространства поиска не связана со значением целевого функционала.

Применение принципа малых вариаций базисного решения для различных методов символьной регрессии позволяет улучшить его возможности при поиске оптимального математического выражения. В работе [11] приведено сравнение и показано преимущество метода вариационного Декартового генетического программирования, использующего принцип малых вариаций базисного решения, перед обычным Декартовым генетическим программированием.

3. Примеры машинного обучения управления

В классической монографии [12] приведено полное решение задачи общего синтеза управления для линейного объекта второго порядка

$$\begin{aligned}\dot{x}_1 &= x_2, \\ \dot{x}_2 &= u,\end{aligned}$$

где

$$-1 \leq u \leq 1.$$

В задаче необходимо попасть из любой точки пространства состояний в терминальную точку начала координат

$$\mathbf{x}^f = [0 \ 0]^T,$$

за минимальное время

$$J = t_f \rightarrow \min.$$

В качестве решения задачи было получено следующее управление

$$u = 1 - 2\vartheta(y(x_1, x_2)),$$

где $\vartheta(a)$ - функция Хэвисайда

$$\vartheta(a) = \begin{cases} 1, & \text{если } a \geq 0 \\ 0 & \text{- иначе} \end{cases},$$
$$y(x_1, x_2) = \begin{cases} 1, & \text{если } x_1 \geq 0, x_2 \geq 0 \\ x_1 - 0.5x_2^2 & \text{- если } x_1 \geq 0, x_2 < 0 \\ -1, & \text{если } x_1 < 0, x_2 \leq 0 \\ x_1 + 0.5x_2^2 & \text{- если } x_1 < 0, x_2 > 0 \end{cases}.$$

При этом вставленные в решение операторы условия были результатом анализа множества экстремалей, полученных при решении системы с предельными значениями управления.

Для решения этой же задачи методом сетевого оператора было выбрано 20 начальных условий и функционал представлял собой сумму точности и времени попадания в терминальное состояние из всех начальных условий

$$J = \sum_{i=1}^{20} \left(t_{f,i} + \|\mathbf{x}^f - \mathbf{x}(t_{f,i}, \mathbf{x}^{0,i})\| \right) \rightarrow \min,$$

где $\mathbf{x}(t, \mathbf{x}^{0,i})$ - частное решение системы дифференциальных уравнений модели из начального условия $\mathbf{x}^{0,i}$, $i = 1, \dots, 20$.

В результате было получено следующее решение

$$u = h(\mathbf{x}^f - \mathbf{x}) = \begin{cases} 1, & \text{если } \tilde{u} \geq 1 \\ -1, & \text{если } \tilde{u} \leq -1 \\ \tilde{u} & \text{иначе} \end{cases},$$

где

$$\begin{aligned} \tilde{u} &= \arctan(A) + B^{-1} + C^3, \\ A &= B + \arctan \left(q_1 \operatorname{sgn}(x_1^f - x_1) \sqrt{|x_1^f - x_1|} \right), \\ B &= q_1(C + \operatorname{sgn}(C) \exp(-|C|)), \\ C &= q_2(x_2^f - x_2) + (C + \operatorname{sgn}(C) \operatorname{sgn}(x_1^f - x_1) \sqrt{|x_1^f - x_1|}), \end{aligned}$$

$$q_1 = 11.3342, \quad q_2 = 8.1389.$$

Полученная в результате машинного обучения функция управления обеспечивает построение поля экстремалей, совпадающее с высокой точностью с аналитическим решением из всех начальных условий, в том числе и не входящих в обучающую выборку.

Рассмотрим задачу остановки между двумя препятствиями автомобильного робота

$$\begin{aligned} \dot{x}_1 &= u_1 \cos(x_3), \\ \dot{x}_2 &= u_1 \sin(x_3), \\ \dot{x}_3 &= u_1 L^{-1} \tan(u_2), \end{aligned}$$

где $L = 4$,

$$\begin{aligned} -5 &= u_1^- \leq u_1 \leq u_1^+ 5, \\ -1 &= u_2^- \leq u_2 \leq u_2^+ = 1. \end{aligned}$$

Робот имеет четыре угла и фазовые ограничения также представляли собой два прямоугольных препятствия, поэтому проверка условий нарушения фазовых ограничений содержала 16 неравенств.

При синтезе функции управления использовалось четыре начальных условия. Функционал качества включал сумму времени попадания в терминальное состояние, точность попадания и штраф за нарушение фазовых ограничений из каждого начального состояния.

В результате машинного обучения управления методом вариационного генетического программирования была получена следующая функция управления

$$u_i = \begin{cases} u_i^+, & \text{если } \tilde{u}_i \geq u_i^+ \\ u_i^-, & \text{если } \tilde{u}_i \leq u_i^- \\ \tilde{u}_i - \text{ иначе} \end{cases}, \quad i = 1, 2,$$

где

$$\tilde{u}_1 = q_1 \Delta_1,$$

$$\tilde{u}_2 = f_{3,1}((\arctan(\Delta_2) - \arctan^3(\Delta_2))^{-1}, \max\{0, \text{sgn}(\Delta_2), q_1\}) \text{sgn}(R) \sqrt{|R|},$$

$$R = f_{3,2}(\vartheta(\min\{q_2, \Delta_3\}), \text{sgn}(\Delta_3) \sqrt{|\mu(\Delta_3)|} + \Delta_3, P),$$

$$P = f_{3,3}(\Delta_3, q_3 \vartheta(\max\{q_4, \Delta_1^2\} + q_1)),$$

$$f_{3,1}(a, b, c) = \begin{cases} c, & \text{если } a \leq b \\ -c - \text{ иначе} \end{cases},$$

$$f_{3,2}(a, b, c) = \begin{cases} b, & \text{если } (a \leq b) \wedge (b < c) \\ a, & \text{если } (a > b) \wedge (a > c) \\ c - \text{ иначе} \end{cases},$$

$$f_{3,3}(a, b, c) = \begin{cases} b - a, & \text{если } |a - b| < |a - c| \\ c - a - \text{ иначе} \end{cases},$$

$$\mu(a) = \begin{cases} a, & \text{если } |a| < 1 \\ \text{sgn}(a) - \text{ иначе} \end{cases},$$

$\Delta_1 = x_1^f - x_1$, $\Delta_2 = x_2^f - x_2$, $\Delta_3 = x_3^f - x_3$, $q_1 = 15.3208$, $q_2 = 13.6169$, $q_3 = 1.5159$, $q_4 = 2.8013$.

Полученное в результате машинного обучения управление переводит мобильный робот из всех начальных условий в терминальное без нарушения ограничений за минимальное время.

Заключение

Основной отличительной особенностью систем интеллектуального управления является наличие в их программных реализациях большого количества операторов условия. Большинство математических методов не получают в качестве решения оператор условия. Методы символьной регрессии позволяют находить структуру и параметры математического выражения функции управления с помощью эволюционного алгоритма. Методы обеспечивают встраивание в функцию управления необходимое количество операторов условия, поэтому методы символьной регрессии следует использовать для синтеза или машинного обучения интеллектуальных систем управления. В дальнейших исследованиях необходимо совершенствовать алгоритмы машинного обучения управления методами символьной регрессии.

Список литературы

- [1] Болтянский В.Г., *Математические методы оптимального управления*, «Наука», Москва, 1969, 408 с.
- [2] Koza, J.R., *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press, Cambridge, MA, USA, 1992, 840 pp.
- [3] Koza, J.R., Keane, M.A., Streeter, M.J., Mydlowec, W., Yu, J., Lanza, G., *Genetic Programming IV: routine human-competitive intelligence.*, Springer, New York. USA, 2005, 590 pp.
- [4] Duriez, T., Brunton, S.L., Noack, B.R., *Machine Learning Control – Taming Nonlinear Dynamics and Turbulence*, Springer, Switzerland, 2017, 212 pp.
- [5] Дивеев А.И., *Численные методы решения задачи синтеза управления*, РУДН, Москва, 2019, 192 с.
- [6] Diveev, A.I., Shmalko E.Yu., *Machine Learning Control by Symbolic Regression*, Springer, Switzerland, 2021, 158 pp.
- [7] Miller, J., Thomson, P., “Cartesian Genetic Programming”, *Lecture Notes in Computer Science. Proceedings of the European Conference on Genetic Programming (EuroGP2000)*, **1802** (2000), 121–132
- [8] Дивеев А.И., Софронова Е.А., *Метод сетевого оператора и его применение в задачах управления*, РУДН, Москва, 2012, 182 с.

- [9] Diveev, A.I., “Small Variations of Basic Solution Method for Non-numerical Optimization jour IFAC-PapersOnLine”, **48:25** (2015), 028—033
- [10] Diveev, A.I., Sofronova, E.A., “Universal Approach to Solution of Optimization Problems by Symbolic Regression”, *Applied Sciences*, **11:5081** (2021)
- [11] Diveev, A.I., Shmalko E.Yu., “Machine-Made Synthesis of Stabilization System by Modified Cartesian Genetic Programming”, *IEEE Transactions on Cybernetics, Early Access Article*, 2020
- [12] Понтрягин Л.С., Болтянский В.Г., Гамкрелидзе Р.В., Мищенко Е.Ф., *Математическая теория оптимальных процессов*, 4, Наука, Москва, 1983, 392 с.

Machine Learning of Intelligent Control Systems

Diveev A.I.

Machine learning intelligent control systems by symbolic regression methods is considered. Symbolic regression allows to find mathematical expressions for various problems where it is necessary to find structure and parameters of unknown multidimensional function. The search for an unknown function is carried out by a genetic algorithm in code space of symbolic regression method. Functions containing condition operators that are mandatory component of intelligent control systems programs.

Keywords: symbolic regression, control synthesis, machine learning, optimal control.

References

- [1] Boltyanskii, V.G., *Mathematical Methods of Optimal Control*, “Science”, Moscow, 1969, 408 (in Russian) с.
- [2] Koza, J.R., *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press, Cambridge, MA, USA, 1992, 840 pp.
- [3] Koza, J.R., Keane, M.A., Streeter, M.J., Mydlowec, W., Yu, J., Lanza, G., *Genetic Programming IV: routine human-competitive intelligence.*, Springer, New York. USA, 2005, 590 pp.

- [4] Duriez, T., Brunton, S.L., Noack, B.R., *Machine Learning Control – Taming Nonlinear Dynamics and Turbulence*, Springer, Switzerland, 2017, 212 pp.
- [5] Diveev, A.I., *Numerical Methods for Solution of the Control Synthesis Problem*, Peoples' Friendship University of Russia, Moscow, 2019, 192 (in Russian) c.
- [6] Diveev, A.I., Shmalko E.Yu., *Machine Learning Control by Symbolic Regression*, Springer, Switzerland, 2021, 158 pp.
- [7] Miller, J., Thomson, P., “Cartesian Genetic Programming”, *Lecture Notes in Computer Science. Proceedings of the European Conference on Genetic Programming (EuroGP2000)*, **1802** (2000), 121–132
- [8] Diveev, A.I., Sofronova, E.A., *Network Operator Method and its Application for Problems of Control*, Peoples' Friendship University of Russia, Moscow, 2012, 182 (in Russian) c.
- [9] Diveev, A.I., “Small Variations of Basic Solution Method for Non-numerical Optimization jour IFAC-PapersOnLine”, **48:25** (2015), 028–033
- [10] Diveev, A.I., Sofronova, E.A., “Universal Approach to Solution of Optimization Problems by Symbolic Regression”, *Applied Sciences*, **11:5081** (2021)
- [11] Diveev, A.I., Shmalko E.Yu., “Machine-Made Synthesis of Stabilization System by Modified Cartesian Genetic Programming”, *IEEE Transactions on Cybernetics, Early Access Article*, 2020
- [12] Pontryagin, L.S., Boltyanskii V.G., Gamkrelidze R.V., Mischenko, E.F., *Mathematical Theory of Optimal Processes*, 4 (in Russian), “Science”, Москва, 1983, 392 c.