



DOI 10.22363/2312-8143-2017-18-1-125-134

МЕТОД БИНАРНОГО ГЕНЕТИЧЕСКОГО ПРОГРАММИРОВАНИЯ ДЛЯ ПОИСКА МАТЕМАТИЧЕСКОГО ВЫРАЖЕНИЯ

А. И. Дивеев^{1,2}, Е. М. Ломакова²

¹ Федеральный исследовательский центр «Информатика и управление» РАН

ул. Вавилова, 44, Москва, Россия, 119333

² Инженерная академия

Российский университет дружбы народов

ул. Миклухо-Маклая, 6, Москва, Россия, 117198

В известных методах символьной регрессии, которые производят поиск решения с помощью генетического алгоритма, существует проблема выполнения операции скрещивания. Все методы либо выполняют скрещивание в определенных точках, как метод генетического программирования, либо корректируют после скрещивания, либо используют избыточные элементы кода. В данной работе представлен новый метод символьной регрессии, основанный на бинарном дереве вычислений. Метод не имеет проблем с операцией скрещивания, включает небольшое количество избыточных элементов кода, но позволяет осуществлять поиск математических выражений, состоящих из функций только с одним или двумя аргументами.

Ключевые слова: символьная регрессия, генетическое программирование, генетический алгоритм

Введение

Методы символьной регрессии могут осуществлять поиск структуры математического выражения. Многие важные задачи требуют нахождения оптимального математического выражения, в том числе задачи обработки экспериментальных данных для прогнозирования и поиска общих факторов, идентификации математической модели и синтеза управления.

Все методы символьной регрессии кодируют математические выражения и осуществляют поиск на пространстве кодов с помощью эволюционного алгоритма. Наиболее подходящим для поиска решения в методах символьной регрессии является генетический алгоритм. Он не использует арифметические операции для модификации возможных решений. Основными операциями генетического алгоритма являются операции скрещивания и мутация. Операция скрещивания генетического алгоритма зависит от вида кода. Генетическое программирование [1] делает операцию скрещивания в определенных точках корректных кодов для того, чтобы новый код после скрещивания был правильным и соответствовал некоторому возможному решению. Методы грамматической эволюции [2] и аналитического программирования [3] осуществляют скрещивание в любых точках, как генетический алгоритм для численных задач. После применения операции

скрещивания в данных методах иногда получается неправильный код. Исправление кода требует дополнительных затрат времени. Методы сетевого оператора [4], декартово генетического программирования [5] и метод матриц синтаксического разбора [6] всегда применяют корректную операцию скрещивания, но эти методы символьной регрессии используют избыточные элементы в коде.

В работе представлен новый метод символьной регрессии, метод бинарного аналитического программирования. Метод кодирует математические выражения только в виде суперпозиции функций с одним или двумя аргументами. Для корректного выполнения операции скрещивания метод также включает в себя избыточные элементы кода, тождественную функцию с одним аргументом и единичными элементами для функций с двумя аргументами. Эти элементы, как и нули в позиционной записи чисел, также позволяют построить правильный код математического выражения, в котором чередуются коды функций с различным числом аргументов.

Бинарное аналитическое программирование

Для того, чтобы составить код бинарного аналитического программирования, мы используем следующие базовые наборы:

- множество аргументов математического выражения

$$F_0 = (q_1, \dots, q_P, x_1, \dots, x_N); \quad (1)$$

- множество функций с одним аргументом

$$F_1 = (f_{1,1}(z) = z, f_{1,2}(z), \dots, f_{1,R}(z)); \quad (2)$$

- множество функций с двумя аргументами

$$F_2 = (f_{2,1}(z_1, z_2), \dots, f_{2,S}(z_1, z_2)); \quad (3)$$

- множество единичных элементов для функций с двумя аргументами

$$E_2 = (e_1, \dots, e_M). \quad (4)$$

Набор функций с одним аргументом должен включать в себя тождественные функции.

$$f_{1,1}(z) = z. \quad (5)$$

Каждая функция с двумя аргументами множества (3), $\forall f_{2,i}(z_1, z_2) \in F_2$, имеет единичный элемент из множества (4), $\exists e_j \in E_2$,

$$f_{2,i}(e_j, z_2) = z_2, f_{2,i}(z_1, e_j) = z_1, i \in \{1, \dots, S\}, j \in \{1, \dots, M\}. \quad (6)$$

Чтобы сгенерировать код бинарного аналитического программирования, мы объединяем в одно упорядоченное множество аргументов математического выражения (1), множество единичных элементов (4) и множество функций с двумя аргументами (3)

$$\begin{aligned}
 F &= (f_1 = q_1, \dots, f_p = q_p, f_{p+1} = x_1, \dots, f_{p+N} = x_N, \\
 &f_{p+N+1} = e_1, \dots, f_{p+N+M} = e_M, \\
 &f_{p+N+M+1} = f_{2,1}(z_1, z_2), \dots, f_{p+N+M+S} = f_{2,S}(z_1, z_2)).
 \end{aligned} \tag{7}$$

Запишем математическое выражение в виде суперпозиции вложенных функций и аргументов математического выражения

$$y = f_{\alpha_1}(f_{\alpha_2}(\dots f_{\alpha_K} \dots)) = f_{\alpha_1} \circ f_{\alpha_2} \circ \dots \circ f_{\alpha_K}, \tag{8}$$

где $f_{\alpha_i} \in F_0 \cup F_1 \cup F_2, i = 1, \dots, K$.

Для получения кода бинарного генетического программирования суперпозиция функций (8) должна удовлетворять следующим условиям:

- а) аргументами функций с двумя аргументами являются только функции с одним аргументом;
- б) аргументами функций с одним аргументом являются аргументы математических выражений или функции с двумя аргументами или их единичные аргументы;
- с) первой функцией является функция с двумя аргументами.

Если суперпозиция не удовлетворяет условиям кодирования, мы исправляем это, добавив тождественную функцию и / или функцию с двумя параметрами, с единицей в качестве одного из аргументов.

Пусть условие а) не выполнено. Аргументом функции с двумя аргументами является тоже функция с двумя аргументами. Добавим функцию тождества (5) в математическое выражение

$$f_{2,k}(f_{2,l}(z_1, z_2), z_3) = f_{2,k}(f_{1,1}(f_{2,l}(z_1, z_2)), z_3). \tag{9}$$

Аргументом функции с двумя аргументами является аргумент математического выражения. Мы снова добавим функцию тождества в математическое выражение.

$$f_{2,k}(z_1, z_2) = f_{2,k}(f_{1,1}(z_1), z_2), \tag{10}$$

где $z_1 \in F_0$.

Пусть условие б) не выполнено. Аргументом функции с одним аргументом является тоже функция с одним аргументом. Добавим функцию с двумя аргументами и с единицей в качестве одного из аргументов в математическом выражении.

$$f_{1,k}(f_{1,l}(z) = f_{1,k}(f_{2,m}(f_{1,l}(z), f_{1,1}(e_r))), \tag{11}$$

где $e_r \in E_2, f_{2,m}(z, e_r) = z$.

Пусть условие с) не выполнено. Функцией суперпозиции (8) является функция с одним аргументом. Добавим функцию с двумя параметрами, взяв в качестве одного из аргументов единичный элемент, в начало математического выражения

$$f_{1,k}(z) = f_{2,m}(f_{1,k}(z), f_{1,1}(e_r)). \tag{12}$$

После коррекции суперпозиции (8) мы получаем суперпозицию с нечетным количеством элементов и чередующихся элементов из множеств (7) и (2)

$$y = f_{a_1} \circ f_{a_2} \circ \dots \circ f_{a_L}, \quad (13)$$

где $f_{a_{2i+1}} \in F, f_{a_{2i}} \in F_1, i = 0, \dots, (L-1)/2$.

Для получения кода математического выражения мы заменим элементы суперпозиции (13) на их серийные номера в соответствующих наборах F и F_1

$$A = (a_1, \dots, a_L), \quad (14)$$

где $a_{2i+1} = v$, если $f_{a_{2i+1}} = f_v \in F, i = 0, \dots, (L-1)/2$, (15)

$$a_{2i} = m, \text{ если } f_{a_{2i}} = f_{1,m} \in F_1, i = 0, \dots, (L-1)/2. \quad (16)$$

Бинарное дерево графа имеет количество листьев на единицу больше числа узлов. Узлы и листья бинарного дерева графа математического выражения (14) соответствуют элементам множества (7). Если серийный номер элемента является не более чем объединенным числом параметров, переменных и единичных элементов в $F, a_{2i+1} \leq P + N + M, i \in \{0, \dots, (L-1)/2\}$, то элемент множества находится на листе бинарного дерева графа. В противном случае $a_{2i+1} > P + N + M, i \in \{0, \dots, (L-1)/2\}$, а элемент расположен в листе двоичного дерева графа.

Пусть количество узлов σ_n и листьев σ_l в коде (14) математического выражения рассчитываются по формулам

$$\sigma_n = \sum_{i=0}^{(L-1)/2} n_i, \quad (17)$$

$$\sigma_l = \sum_{i=0}^{(L-1)/2} l_i, \quad (18)$$

где

$$n_i = \begin{cases} 1, & \text{если } a_{2i+1} > P + N + M \\ 0 & \text{— иначе} \end{cases}, \quad (19)$$

$$l_i = \begin{cases} 1, & \text{если } a_{2i+1} \leq P + N + M \\ 0 & \text{— иначе} \end{cases}. \quad (20)$$

Для корректного кода математического выражения справедливо уравнение

$$\sigma_l - \sigma_n = 1. \quad (21)$$

Рассмотрим операцию скрещивания для кодов бинарного аналитического программирования. Для того чтобы реализовать функцию операции скрещивания,

мы совершаем обмен подмножеств кодов. Каждое подмножество соответствует коду математического выражения. Точки для операция скрещивания определяются в нечетных позициях кодов.

Определение подмножеств кодов для операция скрещивания является важной задачей.

Подмножество кода (14) математического выражения всегда начинается с нечетного порядкового элемента r и имеет нечетное число элементов k

$$A(r, k) = (a_r, a_{r+1}, \dots, a_{r+k}), \quad (22)$$

где $r + k \leq L$.

Количество узлов и листьев в подмножестве кода математического выражения рассчитывается по формулам

$$\sigma_n(r, k) = \sum_{i=(r-1)/2}^{(k-1)/2} n_i, \quad (23)$$

$$\sigma_l(r, k) = \sum_{i=(r-1)/2}^{(k-1)/2} l_i, \quad (24)$$

где n_i и l_i определяются выражениями (19), (20).

Подмножество $A(r, k)$ кода соответствует математическому выражению, когда условие (21) справедливо для минимального количества элементов в нем

$$\min_k \{\sigma_l(r, k) - \sigma_n(r, k) = 1\}. \quad (25)$$

Для операции скрещивания мы выбираем два кода для математических выражений

$$A_\alpha = (a_{\alpha,1}, \dots, a_{\alpha,L_\alpha}), \quad (26)$$

$$A_\beta = (a_{\beta,1}, \dots, a_{\beta,L_\beta}). \quad (27)$$

Нечетные позиции для точек операции скрещивания определяются случайным образом $r_\alpha \in \{1, \dots, L_\alpha\}$, $r_\beta \in \{1, \dots, L_\beta\}$.

Определим подмножества кодов для математических выражений с помощью уравнения (25)

$$A_\alpha(r_\alpha, k_\alpha) = (a_{\alpha,r_\alpha}, \dots, a_{\alpha,r_\alpha+k_\alpha}), \quad (28)$$

$$A_\beta(r_\beta, k_\beta) = (a_{\beta,r_\beta}, \dots, a_{\beta,r_\beta+k_\beta}), \quad (29)$$

где переменные k_α и k_β удовлетворяют следующие выражения

$$\min_{k_\alpha} \{\sigma_{\alpha,l}(r_\alpha, k_\alpha) - \sigma_n(r_\alpha, k_\alpha) = 1\}, \quad (30)$$

$$\min_{k_\beta} \{ \sigma_{\beta,l}(r_\beta, k_\beta) - \sigma_n(r_\beta, k_\beta) = 1 \}. \quad (31)$$

Поменяем подмножества кодов математических выражений и получим коды для новых математических выражений

$$\tilde{A}_\alpha = (a_{\alpha,1}, \dots, a_{\alpha,r_\alpha-1}, a_{\beta,r_\beta}, \dots, a_{\beta,r_\beta+k_\beta}, a_{\alpha,r_\alpha+k_\alpha+1}, \dots, a_{\alpha,r_\alpha+L_\alpha}), \quad (32)$$

$$\tilde{A}_\beta = (a_{\beta,1}, \dots, a_{\beta,r_\beta-1}, a_{\alpha,r_\alpha}, \dots, a_{\alpha,r_\alpha+k_\alpha}, a_{\beta,r_\beta+k_\beta+1}, \dots, a_{\beta,r_\beta+L_\beta}). \quad (33)$$

Рассмотрим пример. Пусть это множество двух математических выражений

$$y_1 = \sin(q_1 x_1) + \begin{cases} \cos(q_2 x_2 + q_3), & \text{если } q_1 x_1 \leq q_2 x_2 \\ \exp(-q_2 x_2) & \text{— иначе} \end{cases},$$

$$y_2 = \begin{cases} \exp(-q_2 x_2), & \text{если } \cos(q_2 x_2 + q_3) \leq \sin(q_1 x_1 + q_3) \\ \exp(-q_1 x_1) & \text{— иначе} \end{cases}.$$

Составим коды бинарного генетического программирования для математических выражений y_1 и y_2 .

Будем использовать следующие базовые множества

$$F_0 = (x_1, x_2, q_1, q_2, q_3),$$

$$F_1 = (f_{1,1}(z) = z, f_{1,2}(z) = \sin(z), f_{1,3}(z) = \cos(z), f_{1,4}(z) = \exp(z),$$

$$f_{1,5}(z) = -z, f_{1,6}(z) = \vartheta(z), f_{1,7}(z) = 1 - \vartheta(z)),$$

$$F_2 = (f_{2,1}(z_1, z_2) = z_1 + z_2, f_{2,2}(z_1, z_2) = z_1 z_2),$$

$$E_2 = (0, 1),$$

где

$$\vartheta(z) = \begin{cases} 1, & \text{если } z > 0 \\ 0 & \text{— иначе} \end{cases}.$$

Объединим множества F_0 , E_2 и F_2

$$F = (f_1 = x_1, f_2 = x_2, f_3 = q_1, f_4 = q_2, f_5 = q_3, f_6 = 0, f_7 = 1, f_8 = z_1 + z_2, f_9 = z_1 z_2).$$

Наши математические выражения y_1 , y_2 включают в себя функцию с тремя аргументами

$$f_{3,1}(z_1, z_2, z_3) = \begin{cases} z_2, & \text{если } z_1 \leq 0 \\ z_3 & \text{— иначе} \end{cases}.$$

Мы можем представить эти функции с помощью некоторых функций с одним и двумя аргументами

$$\begin{aligned} f_{3,1}(z_1, z_2, z_3) &= (1 - \vartheta(z_1))z_2 + \vartheta(z_1)z_3 = \\ &= f_{2,2}(1 - \vartheta(z_1), z_2) + f_{2,2}(\vartheta(z_1), z_3) = f_{2,2}(f_{1,7}(z_1), z_2) + f_{2,2}(f_{1,6}(z_1), z_3) = \\ &= f_{2,1}(f_{2,2}(f_{1,7}(z_1), z_2), f_{2,2}(f_{1,6}(z_1), z_3)). \end{aligned}$$

Представим математические выражения y_1, y_2 элементами множеств F_0, F_1, F_2

$$\begin{aligned} y_1 &= f_{2,1}(f_{1,2}(f_{2,2}(q_1, x_1)), f_{2,1}(f_{2,2}(f_{1,7}(f_{2,1}(f_{2,2}(q_1, x_1), \\ & f_{1,5}(f_{2,2}(q_2, x_2))))), f_{1,3}(f_{2,1}(f_{2,2}(q_2, x_2), q_3))), \\ & f_{2,2}(f_{1,6}(f_{2,1}(f_{2,2}(q_1, x_1), f_{1,5}(f_{2,2}(q_2, x_2))))), \\ & f_{1,4}(f_{1,5}(f_{2,2}(q_2, x_2))))). \end{aligned}$$

$$\begin{aligned} y_2 &= f_{2,1}(f_{2,2}(f_{1,7}(f_{2,1}(f_{1,3}(f_{2,1}(f_{2,2}(q_2, x_2)), q_3)), f_{1,5}(f_{1,2}(f_{2,1}(f_{2,2}(q_1, x_1), \\ & f_{1,4}(f_{1,5}(f_{2,2}(q_2, x_2))))), f_{2,2}(f_{1,6}(f_{2,1}(f_{1,3}(f_{2,1}(f_{2,2}(q_2, x_2)), q_3))), \\ & f_{1,5}(f_{1,2}(f_{2,1}(f_{2,2}(q_1, x_1), q_3))))), f_{1,4}(f_{1,5}(f_{2,2}(q_1, x_1))))). \end{aligned}$$

Полученные записи математических выражений не удовлетворяют свойствам a, b, c . Исправим оба выражения в соответствии с формулами (9)—(12)

$$\begin{aligned} y_1 &= f_{2,1} \circ f_{1,2} \circ f_{2,2} \circ f_{1,1} \circ q_1 \circ f_{1,1} \circ x_1 \circ f_{1,1} \circ f_{2,1} \circ f_{1,1} \circ f_{2,2} \circ f_{1,7} \circ \\ & f_{2,1} \circ f_{1,1} \circ f_{2,2} \circ f_{1,1} \circ q_1 \circ f_{1,1} \circ x_1 \circ f_{1,5} \circ f_{2,2} \circ f_{1,1} \circ q_2 \circ f_{1,1} \circ \\ & x_2 \circ f_{1,3} \circ f_{2,1} \circ f_{1,1} \circ f_{2,2} \circ f_{1,1} \circ q_2 \circ f_{1,1} \circ x_2 \circ f_{1,1} \circ q_3 \circ f_{1,1} \circ f_{2,2} \circ \\ & f_{1,6} \circ f_{2,1} \circ f_{1,1} \circ f_{2,2} \circ f_{1,1} \circ q_1 \circ f_{1,1} \circ x_1 \circ f_{1,5} \circ f_{2,2} \circ f_{1,1} \circ q_2 \circ \\ & f_{1,1} \circ x_2 \circ f_{1,4} \circ f_{2,1} \circ f_{1,1} \circ 0 \circ f_{1,5} \circ f_{2,2} \circ f_{1,1} \circ q_2 \circ f_{1,1} \circ x_2, \end{aligned}$$

$$\begin{aligned} y_2 &= f_{2,1} \circ f_{1,1} \circ f_{2,2} \circ f_{1,7} \circ f_{2,1} \circ f_{1,3} \circ f_{2,1} \circ f_{1,1} \circ f_{2,2} \circ f_{1,1} \circ q_2 \circ f_{1,1} \circ \\ & x_2 \circ f_{1,1} \circ q_3 \circ f_{1,5} \circ f_{2,1} \circ f_{1,1} \circ 0 \circ f_{1,2} \circ f_{2,1} \circ f_{1,1} \circ f_{2,2} \circ f_{1,1} \circ q_1 \circ \\ & f_{1,1} \circ x_1 \circ f_{1,4} \circ f_{2,1} \circ f_{1,1} \circ 0 \circ f_{1,5} \circ f_{2,2} \circ f_{1,1} \circ q_2 \circ f_{1,1} \circ x_2 \circ f_{1,1} \circ \\ & f_{2,2} \circ f_{1,6} \circ f_{2,1} \circ f_{1,3} \circ f_{2,1} \circ f_{1,1} \circ f_{2,2} \circ f_{1,1} \circ q_2 \circ f_{1,1} \circ x_2 \circ f_{1,1} \circ q_3 \circ \\ & f_{1,5} \circ f_{2,2} \circ f_{1,1} \circ 0 \circ f_{1,2} \circ f_{2,1} \circ f_{1,1} \circ f_{2,2} \circ f_{1,1} \circ q_1 \circ f_{1,1} \circ x_1 \circ f_{1,1} \circ \\ & q_3 \circ f_{1,4} \circ f_{2,2} \circ f_{1,1} \circ 0 \circ f_{1,5} \circ f_{2,2} \circ f_{1,1} \circ q_1 \circ f_{1,1} \circ x_1. \end{aligned}$$

Заменим элементы упорядоченных множеств F_0, F_1, F_2 на номера этих элементов в множествах F_1 и F

$$\begin{aligned} A_1 &= (8, 2, 9, 1, 3, 1, 1, 1, 8, 1, 9, 7, 8, 1, 9, 1, 3, 1, 1, 5, 9, 1, 4, 1, 2, 3, 8, 1, 9, 1, 4, \\ & 1, 2, 1, 5, 1, 9, 6, 8, 1, 9, 1, 3, 1, 1, 5, 9, 1, 4, 1, 2, 4, 8, 1, 6, 5, 9, 1, 4, 1, 2), \end{aligned}$$

$$A_2 = (8, 1, 9, 7, 8, 3, 8, 1, 9, 1, 4, 1, 2, 1, 5, 5, 8, 1, 6, 2, 8, 1, 9, 1, 3, 1, 1, 4, 8, 1, 6, 5, 9, 1, 4, 1, 2, 1, 9, 6, 8, 3, 8, 1, 9, 1, 4, 1, 2, 1, 5, 5, 9, 1, 6, 2, 8, 1, 9, 1, 3, 1, 1, 1, 5, 4, 9, 1, 6, 5, 9, 1, 3, 1, 1).$$

Мы получили коды бинарного аналитического программирования для математических выражений.

Рассмотрим пример операции скрещивания кодов A_1, A_2 .

Определим случайным образом нечетные точки для скрещивания. Пусть $r_1 = 15, r_2 = 21$.

Определим подмножества математической формы формулами (28)—(31), $k_1 = 4, k_2 = 16$,

$$\sigma_{1,l}(15,4) = 0 + 1 + 1 = 2,$$

$$\sigma_{1,n}(15,4) = 1 + 0 + 0 = 1,$$

$$\sigma_{1,l}(15,4) - \sigma_{1,n}(15,4) = 2 - 1 = 1,$$

$$\sigma_{2,l}(21,16) = 0 + 0 + 1 + 1 + 0 + 1 + 0 + 1 + 1 = 5,$$

$$\sigma_{2,n}(21,16) = 1 + 1 + 0 + 0 + 1 + 0 + 1 + 0 + 0 = 4,$$

$$\sigma_{2,l}(21,16) - \sigma_{2,n}(21,16) = 5 - 4 = 1.$$

Мы получили следующие подмножества кодов

$$A_1 = (8, 2, 9, 1, 3, 1, 1, 1, 8, 1, 9, 7, 8, 1, 9, 1, 3, 1, 1, 5, 9, 1, 4, 1, 2, 3, 8, 1, 9, 1, 4, 1, 2, 1, 5, 1, 9, 6, 8, 1, 9, 1, 3, 1, 1, 5, 9, 1, 4, 1, 2, 4, 8, 1, 6, 5, 9, 1, 4, 1, 2),$$

$$A_2 = (8, 1, 9, 7, 8, 3, 8, 1, 9, 1, 4, 1, 2, 1, 5, 5, 8, 1, 6, 2, 8, 1, 9, 1, 3, 1, 1, 4, 8, 1, 6, 5, 9, 1, 4, 1, 2, 1, 9, 6, 8, 3, 8, 1, 9, 1, 4, 1, 2, 1, 5, 5, 9, 1, 6, 2, 8, 1, 9, 1, 3, 1, 1, 1, 5, 4, 9, 1, 6, 5, 9, 1, 3, 1, 1).$$

Новые коды соответствуют новым математическим выражениям

$$\begin{aligned} \tilde{y}_1 &= \sin(q_1 x_1) + (1 - \vartheta(q_1 x_1 + \exp(-q_2 x_2) - q_2 x_2)) \cdot \\ &\quad \cdot \cos(q_2 x_2 + q_3) + \vartheta(q_1 x_1 - q_2 x_2) \exp(-q_2 x_2), \\ \tilde{y}_2 &= (1 - \vartheta(\cos(q_2 x_2 + q_3) - \sin(q_1 x_1))) \cdot \\ &\quad \cdot \vartheta(\cos(q_2 x_2 + q_3) - \sin(q_1 x_1 + q_3)) \exp(-q_1 x_1). \end{aligned}$$

Выводы

Представлен новый метод символьный регрессии, бинарного генетического программирования. Новый метод использует бинарное вычислительное дерево. Метод увеличивает число возможных точек скрещивания и не требует коррекции кода после скрещивания.

© Дивеев А.И., Ломакова Е.М., 2017

БИБЛИОГРАФИЧЕСКИЙ СПИСОК/REFERENCES

- [1] Koza, J.R. Genetic Programming: On the Programming of Computers by Means of Natural Selection. Cambridge, Massachusetts, London, MA: MIT Press, 1992. 819 p.
- [2] O'Neill, M., Ryan, C. Grammatical Evolution. *IEEE Trans. Evol. Comput.* 2001, 5. Pp. 349–358.
- [3] Zelinka, I. Analytic programming by Means of SOMA Algorithm. In Proceedings of 8th International Conference on Soft Computing Mendel 02, 2002, Brno, Czech Republic. Pp. 93–101.
- [4] Diveev, A., Sofronova, E. Application of Network Operator Method for Synthesis of Optimal Structure and Parameters of Automatic Control System. Proc. of 17-th IFAC World Congress, Seoul, 05.07.2008 – 12.07.2008. Pp. 6106–6113.
- [5] Miller, J., Thomson, P. Cartesian Genetic Programming. Proc. EuroGP'2000R 3rd European Conf. Genetic Programming, R. Poli, W. Banzhaf, W.B. Langdon, J.F. Miller, P. Nordin, and Fogarty, T.C. Eds., Edinburgh, Scotland, 2000, vol. 1802. Berlin: Springer-Verlag. Pp. 121–132.
- [6] Luo, C., Zhang, S.-L. Engineering Applications of Artificial Intelligence. 2012, 25. Pp. 1182–1193.

История статьи:

Дата поступления в редакцию: 5 декабря 2016

Дата принятия к печати: 22 января 2017

Для цитирования:

Дивеев А.И., Ломакова Е.М. Метод бинарного генетического программирования для поиска математического выражения // *Вестник Российского университета дружбы народов. Серия: Инженерные исследования*. 2017. Т. 18. № 1. С. 125–134.

Сведения об авторах:

Дивеев Асхат Ибрагимович, доктор технических наук, профессор, заведующий сектором проблем кибернетики, Федеральный исследовательский центр «Информатика и управление», Российская академия наук, профессор департамента механики и мехатроники, Инженерная академия, Российский университет дружбы народов. *Сфера научных интересов*: вычислительные методы для решения задач управления. *Контактная информация*: e-mail: aidiveev@mail.ru.

Ломакова Евгения Михайловна, аспирант, департамент механики и мехатроники, Инженерная академия, Российский университет дружбы народов. *Сфера научных интересов*: Computational methods for problems of control. *Контактная информация*: e-mail: lomakovajm@gmail.com.

METHOD OF BINARY ANALYTIC PROGRAMMING TO LOOK FOR OPTIMAL MATHEMATICAL EXPRESSION

A.I. Diveev¹, E.M. Lomakova²

¹ Federal Research Center “Computer Science and Control” of RAS

Vavilov str., 44, Moscow, Russia, 119333

² Engineering Academy

Peoples' Friendship University of Russia

Miklukho-Maklaya str., 6, Moscow, Russia, 117198

In the known methods of symbolical regression by search of the solution with the help of a genetic algorithm, there is a problem of crossover. Genetic programming performs a crossover only in certain

points. Grammatical evolution often corrects a code after a crossover. Other methods of symbolical regression use excess elements in a code for elimination of this shortcoming. The work presents a new method of symbolic regression on base of binary computing trees. The method has no problems with a crossover. Method use a coding in the form of a set of integer numbers like analytic programming. The work describes the new method and some examples of coding for mathematical expressions.

Key words: symbolic regression, genetic programming, genetic algorithm

Article history:

Received: 5 December 2016

Accepted: 22 January 2017

For citation:

Diveev A.I., Lomakova E.M. (2017) Synthesis of control for group of autonomous robots with phase constraints by multi-layer network operator with priorities. *RUDN Journal of Engineering Researches*, 18(1), 125—134.

Bio Note:

Askhat I. Diveev, Doctor of technical sciences, professor, chief of sector of Cybernetic problems, Federal Research Centre “Computer Science and Control” of Russia Academy of Sciences, professor of department Mechanics and mechatronics, Engineering Academy, Peoples’ Friendship University of Russia (RUDN University). *Research Interests:* Computational methods for problems of control. *Contact information:* e-mail: aidiveev@mail.ru.

Evgenia M. Lomakova, graduate student, department Mechanics and mechatronics, Engineering Academy, Peoples’ Friendship University of Russia (RUDN University). *Research Interests:* Computational methods for problems of control. *Contact information:* e-mail: lomakovajm@gmail.com.