

RESEACH OF NON CLOSED ADAPTIVE HYDROSTATIC SUPPORT WITH INDEPENDENT FLOATING REGULATOR

Construction and optimization principle of design-regime parameters of non closed adaptive hydrostatic support with independent floating regulator of work fluid consumption in comparison with traditional hydrostatic supports with throttle control for guide bearing of knots of heavy metal-cutting engineering tools is described. Results of theoretical and experimental researches of load characteristics and construction parameters, by flexibility criteria, are presented.

Keywords: metal-cutting engineering tool, adaptive hydrostatic support, floating regulator of work fluid consumption.

© Ереско С. П., Шатохин С. С., 2011

УДК 519.68

В. Г. Жуков

АВТОМАТИЗАЦИЯ ПРОЦЕССА ПОСТРОЕНИЯ И ОПТИМИЗАЦИИ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ ЭВОЛЮЦИОННЫМИ АЛГОРИТМАМИ НА ОСНОВЕ ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ*

Рассмотрен синтез эволюционных алгоритмов решения задачи символьной регрессии и оптимизации. Предложены их модификации в рамках разработки интегрированной процедуры автоматизированного построения и оптимизации математических моделей сложных систем и процессов.

Ключевые слова: эволюционные алгоритмы, символьная регрессия, оптимизация.

При решении задачи символьной регрессии методом генетического программирования [1] в начале работы алгоритма часто возникает ситуация, при которой сгенерированные деревья решений с более простой структурой (обычно это линейные выражения) имеют более высокую пригодность, чем деревья решений со сложными структурами, которые обычно оказываются более перспективными с точки зрения генерации решения с заданной ошибкой аппроксимации. Эта проблема особенно часто возникает при решении практических задач, в которых результирующее дерево решений, как правило, представляет собой выражение со сложной структурой. В результате в популяции начинают преобладать простые деревья решений и скорость поиска замедляется, так как набор констант, определенный во множестве термов, фиксирован и в сгенерированных структурах численные коэффициенты подобраны плохо. Таким образом, дерево решений с очень хорошей структурой может иметь ошибку аппроксимации намного больше, чем простая структура, в которой меньше коэффициентов [2].

Рассмотрим пример, наглядно показывающий, как плохо подобранные коэффициенты влияют на пригодность дерева решений (рис. 1).

Пусть задана функция $f(x) = 3 \sin(x)$ на интервале $[-2,5; 2,5]$ и в популяции есть два дерева решений: $p_1(x) = x$ и $p_2(x) = x - x^3$ (рис. 1, а).

Если в качестве ошибки аппроксимации взять, например, евклидово расстояние, то для решений $p_1(x)$ и $p_2(x)$ ошибка составит $d(f(x), p_1(x)) = 7,79$ и $d(f(x), p_2(x)) = 47,21$ соответственно. Очевидно, что дерево решений $p_1(x)$, несмотря на то что оно плохо аппроксимирует функцию $f(x)$, для алгоритма генетического программирования (АГП) оказывается предпочтительнее дерева $p_2(x)$. Однако если в $p_2(x)$ подобрать численные коэффициенты, то после оптимизации дерево решений, например

$$p_2(x) = (2,56 \cdot x + 0,01) - (0,3 \cdot x^3 + 0,01),$$

будет иметь меньшую ошибку аппроксимации: $d(f(x), p'_2(x)) = 1,17$, и теперь алгоритм отдаст предпочтение именно этому дереву решений (рис. 1, б).

Следует отметить, что описанная выше проблема может возникнуть и в конце работы алгоритма, когда структура дерева решений уже сгенерирована и ее элементы доминируют в популяции, но элементы множества термов (константы), входящие в нее, не позволяют достичь требуемой точности аппроксимации.

Алгоритм генетического программирования способен разрешить данную проблему путем интенсивного применения оператора мутации, однако случайный характер этого оператора может оказать и негативное воздействие (например, могут быть потеряны хорошо подобранные коэффициенты или изменена сгенерированная структура).

*Работа выполнена в рамках Федеральной целевой программы «Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2007–2012 годы».

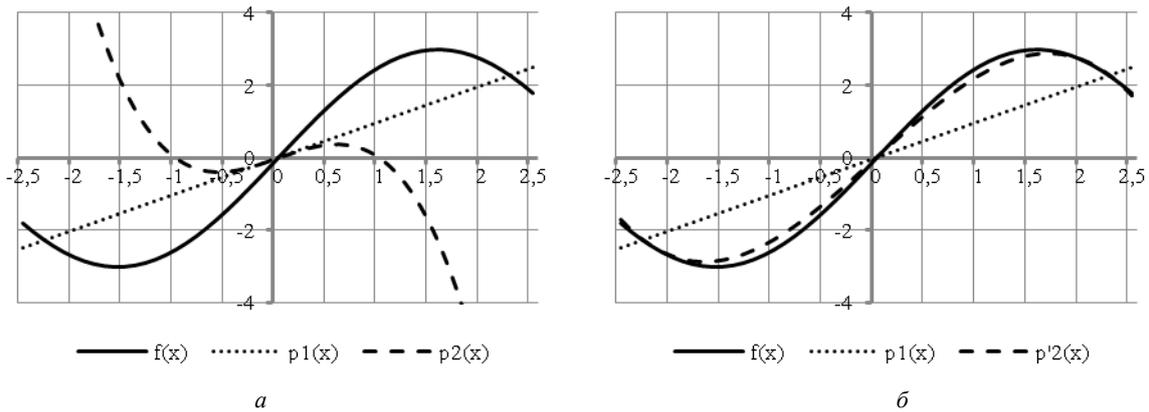


Рис. 1. Пример решений, сгенерированных алгоритмом генетического программирования до (а) и после (б) процедуры оптимизации численных коэффициентов

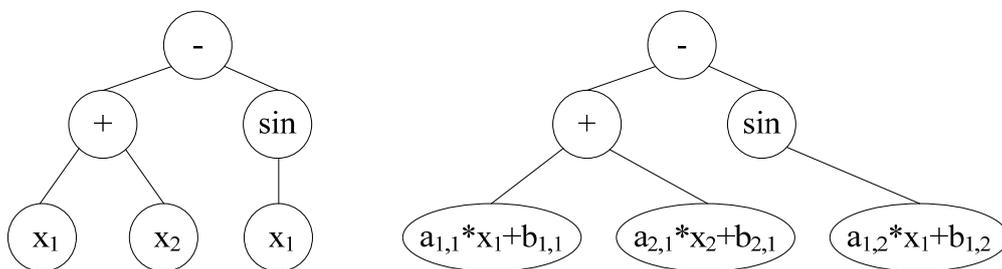


Рис. 2. Пример решения $f(x_1, x_2) = x_1 + x_2 - \sin(x_1)$

Подстройка коэффициентов также возможна за счет появления поддеревьев, содержащих во внешних вершинах только константы. В результате вычисления значения такого поддерева может появиться численный коэффициент, не включенный во множество термов. Тем не менее зачастую алгоритм либо плохо справляется с подбором коэффициентов, либо коэффициенты настраиваются очень долго. Поэтому параметры генерируемых решений необходимо настраивать, что можно сделать только с помощью процедур прямого поиска.

Алгоритм генетического программирования с настройкой значений численных коэффициентов решений, сгенерированных генетическим алгоритмом. Для разрешения проблемы настройки численных коэффициентов в деревьях решений, сгенерированных алгоритмом генетического программирования, может быть предложена следующая модификация: будем представлять вектор независимых входных переменных в виде

$$\bar{X} = (a_{1,i} \cdot x_1 + b_{1,i}, a_{2,i} \cdot x_2 + b_{2,i}, \dots, a_{n,i} \cdot x_n + b_{n,i}),$$

где n – количество переменных; i – порядковый номер вхождения переменной в сгенерированное дерево решений (рис. 2).

Полученные таким образом деревья решений будут представлять собой не готовые структуры, а их структурные шаблоны (функциональные формы). При оценивании решений численные коэффициенты структур настраиваются с помощью внешней оптимизационной процедуры.

При такой модификации алгоритм генетического программирования в случае появления в популяции удачных структур после настройки численных коэффициентов отдаст предпочтение именно этим структурам. В результате при аппроксимации сильно нелинейных зависимостей алгоритм гораздо быстрее перейдет от линейных (или слабо нелинейных) функций к более сложным. При этом очевидно, что количество численных коэффициентов для сложных структур будет велико и при настройке параметров заданной структуры, а функция ошибки аппроксимации, т. е. критерий оптимизации, будет нелинейной и многоэкстремальной.

Как известно, генетический алгоритм (ГА) достаточно эффективно решает сложные задачи оптимизации [3]. Более того, он использует тот же механизм, что и алгоритм генетического программирования. Поэтому в данном случае для настройки численных коэффициентов сгенерированных деревьев решений логично использовать именно генетический алгоритм.

Однако следует отметить, что при применении классического генетического алгоритма в качестве внешней оптимизационной процедуры время работы интегрированной процедуры автоматизированного построения и оптимизации математических моделей сложных систем и процессов значительно возрастает, так как для каждого сгенерированного алгоритмом генетического программирования дерева решений на каждом поколении потребуется настройка численных коэффициентов.

Автором было проведено исследование предложенного алгоритма на базе интегрированной процедуры автоматизированного построения и оптимизации математических моделей сложных систем и процессов для оценки его эффективности. В качестве экспериментальных данных использовались значения функции $\sin(x)$ на интервале $[-3,14; 3,14]$ со случайным шагом и объемом тестовой выборки, равном 100 точкам. В качестве функционального множества рассматривались только арифметические операции сложения, вычитания, умножения и деления, терминальное множество содержало переменную x и набор констант на интервале $[-1;1]$ со случайным шагом.

Вектор параметров алгоритма генетического программирования имел следующий формат: АГП = (размер популяции; метод роста; селекция; метод скрещивания; мутация; вероятность мутации; начальная глубина деревьев; стратегии формирования нового поколения; число поколений) – и был проинициализирован следующими значениями: АГП = (100; метод полного выращивания; пропорциональная; кратный узел; метод роста; 0,5; 3; элитизм; 2 000) [4].

Вектор параметров генетического алгоритма имел следующий формат: ГА = (размер популяции; селекция; стратегия выбора родительской пары; рекомбинация; вероятность мутации; стратегии формирования нового поколения; число поколений) – и был проинициализирован следующими значениями: ГА = (50; турнирная; инбридинг; одноточечная; 0,05; элитизм; 20) [4].

Для наглядности все эксперименты проводились в однопоточном режиме. Обобщенные результаты исследований разработанного алгоритма генетического программирования с настройкой значений численных коэффициентов решений, сгенерированных генетическим алгоритмом, представлены в табл. 1.

В ходе исследования изменялась частота вызова генетического алгоритма в процессе работы алгоритма генетического программирования $K = [1; 8]$ с шагом в единицу. Для каждого K проводилось 20 запусков алгоритма генетического программирования по 2 000 поколений и вычислялись следующие характеристики:

- G – среднее поколение, на котором было получено решение с $E < 1\%$;
- C_n – среднее количество узлов в лучшем индивиде (учитываются только те прогоны, в которых было получено решение с $E < 1\%$);
- N_1 – количество запусков (%), в которых было получено решение с $E > 5\%$;
- N_2 – количество запусков (%), в которых было получено решение с $1\% < E < 5\%$;
- N_3 – количество запусков (%), в которых было получено решение с $E < 1\%$;
- T_{cp} – среднее время работы одного запуска алгоритма генетического программирования с настройкой численных коэффициентов.

На основании данных, приведенных в табл. 1, были построены графики зависимостей $N_3 = f(K)$ (рис. 3) и $T_{cp} = f(K)$ (рис. 4).

Таблица 1

Исследование АГП с настройкой коэффициентов

K	$N_1, \%$	$N_2, \%$	$N_3, \%$	G	C_n	$T_{cp} \cdot 10^3$ мин
1	5	45	50	252	324	4,5
2	5	50	45	347	347	3,6
3	5	50	45	575	321	3,2
4	10	50	40	646	326	2,9
5	5	50	45	837	323	2,7
6	5	55	40	1 265	314	2,4
7	10	50	40	1 655	345	2,1
8	10	55	35	1 823	352	1,8

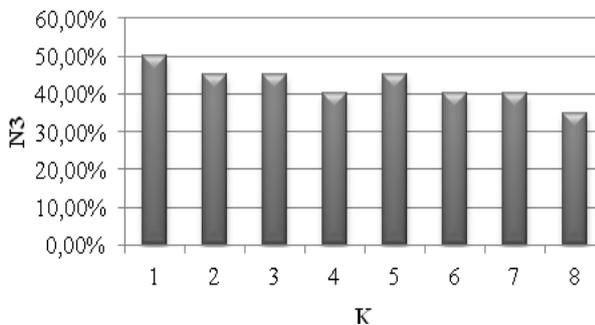


Рис. 3. График зависимости $N_3 = f(K)$

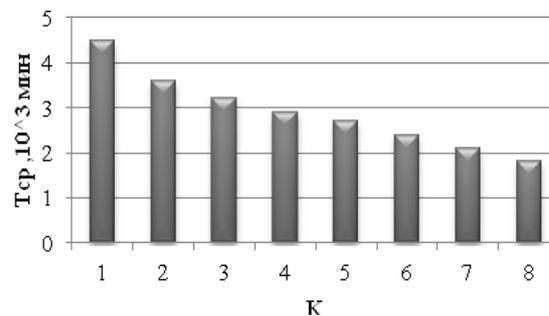


Рис. 4. График зависимости $T_{cp} = f(K)$

Анализ графиков рис. 3 и 4 показывает, что наиболее приемлемым по среднему времени работы алгоритма и количеству запусков, в которых было получено решение с заданной точностью, является значение $K = 5$. Данное значение логически оправданно, так как алгоритму генетического программирования необходим некоторый интервал адаптации для того, чтобы сгенерировать новые структуры, определить их потенциальную пригодность и закрепить их в последующих поколениях.

Еще одной особенностью предложенного интегрированного алгоритма, которая будет оказывать существенное влияние на время и потребляемые ресурсы, является большая размерность решаемой задачи безусловной однокритериальной оптимизации значений численных коэффициентов. Это обусловлено тем, что для сложных структур деревьев решений, особенно в конце работы алгоритма, размерность задачи оптимизации будет, как минимум, не меньше удвоенного количества входных независимых переменных алгоритма генетического программирования. Поэтому с точки зрения потребляемых вычислительных ресурсов проводить оптимизацию численных коэффициентов сгенерированных деревьев решений на каждом поколении алгоритма генетического программирования невыгодно даже в случае применения многопоточных и/или распределенных вычислений.

Однако следует отметить, что разработанный интегрированный алгоритм имеет следующие недостатки:

- при оптимизации численных коэффициентов генетическим алгоритмом не учитывается информация, полученная алгоритмом генетического программирования;

- время работы генетического алгоритма несопоставимо велико по сравнению со временем работы алгоритма генетического программирования даже с учетом введения интервала адаптации.

Таким образом, применение классической реализации генетического алгоритма для оптимизации значений численных коэффициентов с точки зрения потребляемых ресурсов и времени работы является необоснованным и для повышения эффективности требует внесения модификаций.

Алгоритм генетического программирования с настройкой значений численных коэффициентов решений, сгенерированных модифицированным генетическим алгоритмом. Для улучшения эффективности работы генетического алгоритма, применяемого для настройки численных коэффициентов через интервал адаптации, были предложены следующие его модификации [5]:

- переход численных коэффициентов лучших решений алгоритма генетического программирования;
- накопление и использование генетическим алгоритмом статистических данных о пространстве поиска;
- управление ресурсом генетического алгоритма, выделяемого для решения задачи оптимизации.

Рассмотрим предложенные модификации более подробно.

Переход численных коэффициентов лучших решений алгоритма генетического программирования. Если пригодность (под которой будем понимать величину, обратную нормированной ошибке аппроксимации) некоторого i -го дерева решения, сгенерированного алгоритмом генетического программирования, удовлетворяет условию $fitness_i \geq fitness_{cp}$, где $fitness_{cp}$ – средняя пригодность популяции на j -м поколении, то при параметрической оптимизации численные коэффициенты данного решения кодируются и включаются в первую популяцию генетического алгоритма. Тем самым учитывается информация, полученная алгоритмом генетического программирования, и оптимизационный алгоритм начинает свою работу, имея в первой популяции вполне конкретные точки с пригодностью выше средней. Конечно, нельзя утверждать, что такая модификация приведет к нахождению оптимальных численных коэффициентов во всех случаях, но оптимизационный алгоритм достаточно быстро улучшит исходные численные коэффициенты и не позволит алгоритму генетического программирования потерять хорошие с точки зрения пригодности, структуры, которые, как правило, являются нелинейными.

Таким образом, применение данной модификации позволяет алгоритму генетического программирования гораздо быстрее перейти от линейных (или слабо нелинейных) структур к более сложным за меньшее количество поколений (рис. 5).

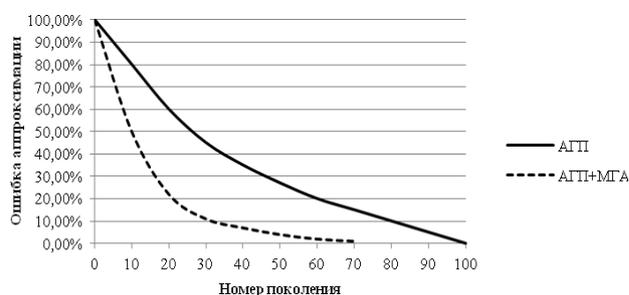


Рис. 5. Пример динамики уменьшения ошибки аппроксимации при применении модифицированного генетического алгоритма

Накопление и использование генетическим алгоритмом статистических данных о пространстве поиска. В ходе параметрической оптимизации генетический алгоритм накапливает и обрабатывает некоторую статистическую информацию о пространстве поиска, однако эта статистика в явном виде отсутствует. Для анализа работы генетического алгоритма предложен следующий способ представления накопленной генетическим алгоритмом статистики.

На каждом поколении генетического алгоритма вычисляется средняя пригодность $fitness_{cp}$. Далее подсчитывается количество единиц в j -м гене k -й хромосомы, удовлетворяющей неравенству $fitness_k \geq fitness_{cp}$. На основе статистических данных определяются вероятности появления единицы в j -м гене: $p_i = M_i / k$,

где M_i – количество единиц в j -м гене; k – количество хромосом в популяции с $fitness_k \geq fitness_{cp}$.

Полученные вероятности используются при формировании хромосомы, значения генов которой инициализируются по следующему правилу (рис. 6):

$$- p = 1, \text{ если } p_i \geq \overline{p_{\max}} = \frac{1}{k_{\max}} \cdot \sum_{i=1}^{k_{\max}} p_{i_{\max}}, \text{ где}$$

$$p_{i_{\max}} \geq 0,5;$$

$$- p = 0, \text{ если } p_i \leq \overline{p_{\min}} = \frac{1}{k_{\min}} \cdot \sum_{i=1}^{k_{\min}} p_{i_{\min}},$$

где $p_{i_{\min}} < 0,5$;

– значение гена инициализируется случайным образом, если $\overline{p_{\min}} < p_i < \overline{p_{\max}}$.

Пусть например, вероятности имеют следующие значения:

№ гена	1	2	3	4	5	6	7	8	9
P	0,54	0,25	0,19	0,8	0,9	0,32	0,65	0,12	0,93

Тогда

$$\begin{aligned} \overline{p_{\max}} &= \frac{1}{5} \cdot \sum_{i=1}^5 p_i^{\max} = \\ &= \frac{1}{5} \cdot (0,54 + 0,8 + 0,9 + 0,65 + 0,93) = 0,764, \end{aligned}$$

$$\begin{aligned} \overline{p_{\min}} &= \frac{1}{5} \cdot \sum_{i=1}^5 p_i^{\min} = \\ &= \frac{1}{4} \cdot (0,25 + 0,19 + 0,32 + 0,12) = 0,22. \end{aligned}$$

В результате будет получена хромосома:

№ гена	1	2	3	4	5	6	7	8	9
P	Rand	Rand	0	1	1	Rand	Rand	0	1

Эта хромосома, созданная на основе статистических данных каждого поколения, искусственно вводится в популяцию следующего поколения. В результате исследований было установлено, что в 85 % случаев пригодность полученной хромосомы выше средней на каждом поколении. Таким образом, данная хромосома задает вектор движения алгоритма и ускоряет его работу за счет распространения значений своих генов на всю популяцию.

Управление ресурсом генетического алгоритма, выделяемого для решения задачи оптимизации. В процессе работы алгоритма генетического программирования целью применения генетического алгоритма с предложенными выше модификациями является увеличение скорости генерирования структуры дерева решения с заданной ошибкой аппроксимации за счет повышения вероятности отбора нелинейных структур путем настройки численных коэффициентов. Но нельзя заранее предугадать, на каком именно поколении будет сгенерирована данная структура, из-за чего применение генетического алгоритма с ресурсом, сопоставимым с ресурсом алгоритма генетического программирования, даже с учетом интервала адаптации является необоснованным. Поэтому было предложено ограничить ресурс генетического алгоритма 20 % от ресурса алгоритма генетического программирования. Однако если генетический алгоритм находит решение с пригодностью, большей чем исходная пригодность этого же решения, найденного с помощью метода генетического программирования до настройки численных коэффициентов, то ресурс генетического алгоритма однократно увеличивается вдвое.

Эффективность модифицированного алгоритма в рамках интегрированной процедуры автоматизированного построения и оптимизации математических моделей сложных систем и процессов проверена на тестовых функциях с усреднением по многим запускам.

Работа гибридного алгоритма с применением предложенных модификаций исследовалась на задаче построения модели функции $\sin(x)$ на интервале $[-3,14; 3,14]$ со случайным шагом и объемом тестовой выборки, равном 100 точкам. В качестве функционального множества использовались арифметические операции сложения, вычитания, умножения и деления, терминальное множество содержало переменную x и набор констант на интервале $[-1;1]$ со случайным шагом.

Полученные результаты представлены в табл. 2, 3 и на рис. 7.

Таким образом, применение модифицированного генетического алгоритма для настройки численных коэффициентов решений, сгенерированных методом генетического программирования, способствовало увеличению эффективности работы метода генетического программирования по показателю N_3 ($E < 1\%$) на 15 % и снижению количества решений с ошибкой аппроксимации больше 5 % до нуля.

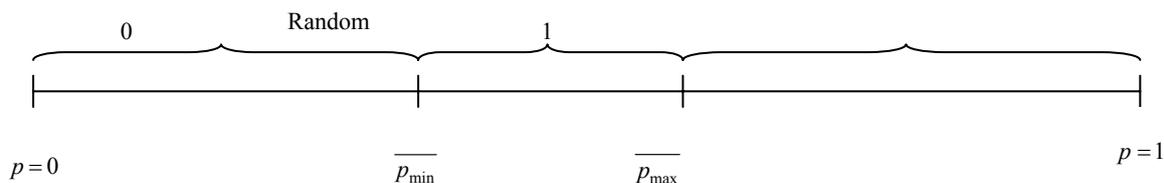


Рис. 6. Правила формирования хромосомы на основе статистических данных

Обобщенные результаты исследований

Модификации	$N_1, \%$	$N_2, \%$	$N_3, \%$	G	C_n
Использование статистических данных ГА	5	50	45	751	361
Управление ресурсом ГА	5	50	45	783	352
Использование статистических данных ГА + управление ресурсом ГА + переход коэффициентов лучших решений АГП	0	45	55	632	314

Сравнительный анализ эффективности алгоритмов

№ п/п	Алгоритм	$N_1, \%$	$N_2, \%$	$N_3, \%$
1	АГП	20	40	40
2	АГП + ГА	5	45	50
3	АГП + ГА + использование статистических данных ГА	5	50	45
4	АГП + ГА + использование статистических данных ГА + Управление ресурсом ГА	5	50	45
5	АГП+ГА + использование статистических данных ГА + управление ресурсом ГА + + переход коэффициентов лучших решений АГП	0	45	55

Синтез модифицированных эволюционных алгоритмов позволил создать гибридный алгоритм решения задачи символьной регрессии и оптимизации в рамках интегрированной процедуры автоматизированного построения и оптимизации математических моделей сложных систем и процессов на основе экспериментальных данных, превосходящий стандартный алгоритм генетического программирования по быстродействию и надежности.

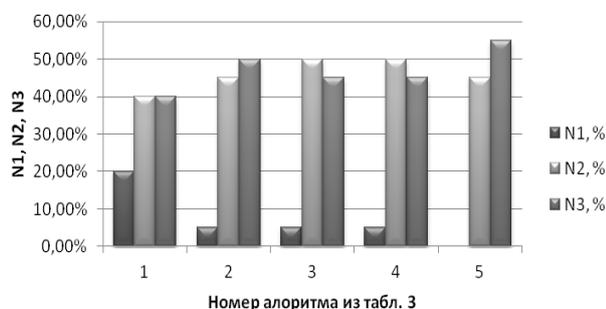


Рис. 7. Сравнительный анализ эффективности стандартных и предложенных алгоритмов

Предложенный алгоритм развивает эволюционную теорию моделирования и оптимизации сложных систем, а также теорию и практику поддержки принятия решений при управлении этими системами.

Библиографические ссылки

1. Koza J. R. Genetic Programming: On Programming Computer by Means of Natural Selection and Genetics. Cambridge, Mass. : MIT Press, 1992.
2. Сопов Е. А. Эволюционные алгоритмы моделирования и оптимизации сложных систем : дис. ... канд. техн. наук. Красноярск, 2004.
3. Holland J. H. Adaptation in Natural and Artificial Systems. Ann Arbor : Univ. of Michigan Press, 1975.
4. Жуков В. Г. Моделирование сложных систем коэволюционным алгоритмом генетического программирования : дис. ... канд. техн. наук. Красноярск, 2006.
5. Жуков В. Г. О решении проблемы настройки численных коэффициентов при решении задачи символьной регрессии методом генетического программирования // Сб. тр. XII Нац. конф. по искусств. интеллекту / Твер. гос. техн. ун-т. Тверь, 2010. С. 311–317.

V. G. Zhukov

AUTOMATION OF CONSTRUCTION AND OPTIMIZATION OF MATHEMATICAL MODELS EVOLUTIONARY ALGORITHMS BASED ON EXPERIMENTAL DATA

The author considers synthesis of evolutionary algorithms for solution of the problem of symbolic regression and optimization, and their proposed modifications, through the development of integrated procedures of automated construction and optimization of mathematical models of complex systems and processes.

Keywords: evolutionary algorithms, symbolic regression, optimization.

© Жуков В. Г., 2011