

О ВЛИЯНИИ ОСОБЕННОСТЕЙ ФИТНЕСС-ФУНКЦИЙ НА СХОДИМОСТЬ ГЕНЕТИЧЕСКОГО АЛГОРИТМА

И.О. Лукьянов, Ф.А. Литвиненко

Рассматриваются адаптивные возможности параллельной версии многопопуляционного генетического алгоритма в зависимости от особенностей определенных классов фитнесс-функций. Предлагаются способы повышения скорости сходимости к оптимальному решению на основе эффективного управления параметрами алгоритма и стратегиями обмена хромосомами-решениями между популяциями. Приводятся результаты компьютерных экспериментов при оптимизации фитнесс-функций с различными соотношениями малозначимых и значимых факторов. Исследуется зависимость скорости сходимости алгоритма при наличии случайного воздействия на значения фитнесс-функций.

Ключевые слова: многопопуляционный генетический алгоритм, малозначимые факторы, компьютерная модель.

Розглядаються адаптивні можливості паралельної версії багатопопуляційного генетичного алгоритму в залежності від особливостей певних класів фітнесс-функцій. Пропонуються способи підвищення швидкості знаходження оптимального рішення на основі ефективного керування параметрами алгоритму і стратегіями обміну хромосомами-рішеннями між популяціями. Наводяться результати комп'ютерних експериментів при оптимізації фітнесс-функцій з різними співвідношеннями малозначущих і значущих факторів. Досліджується залежність швидкості роботи алгоритму при наявності випадкового впливу на значення фітнесс-функцій.

Ключові слова: багатопопуляційний генетичний алгоритм, малозначущі фактори, комп'ютерна модель.

The adaptive capabilities of a parallel version of a multipopulation genetic algorithm are considered depending on the characteristics of certain classes of fitness-functions. Ways are proposed to increase the rate of convergence to the optimal solution based on effective control of algorithm parameters and strategies for the exchange of chromosome-solutions between populations. The results of computer experiments with the optimization of fitness-functions with various ratios of insignificant and significant factors are presented. The dependence of the convergence rate of the algorithm in the presence of a random effect on the values of fitness-functions is studied.

Key words: multipopulation genetic algorithm, insignificant factors, computational model.

Вступление

Задачи имитационного компьютерного моделирования зачастую имеют большую размерность. При разработке сложной модели стараются по возможности учесть все факторы, оказывающие влияние на результат её работы. Однако на практике многие из них лишь незначительно влияют на эффективность моделируемой системы (назовем их малозначимыми). При этом их присутствие в модели существенно увеличивает ее размерность, а значит и время ее «прогона». Кроме того, некоторые реакции имитационной модели, как правило, имеют случайный характер, что также усложняет процесс её оптимизации.

Эволюционные алгоритмы показывают высокую эффективность при решении многих оптимизационных задач большой размерности и, в частности, при реализации оптимизационно-имитационных экспериментов [1–3]. В работах [4–6] подробно описана параллельная версия многопопуляционного генетического алгоритма, который оптимизирован для решения некоторого класса задач с единственным экстремумом и большим пространством значений факторов. Одной из особенностей применения генетического алгоритма при решении этих задач является преобладание операций скрещивания над операциями мутации, поскольку последние с большей вероятностью ухудшают значение фитнесс-функции, нежели улучшают его, что существенно замедляет нахождение оптимального решения.

Цель данной работы – исследование влияния малозначимых и случайных факторов на эффективность работы описанного в [4–6] многопопуляционного генетического алгоритма.

Описание тестовых задач

В качестве тестовой задачи в работах [4–6] выбрана следующая: используя генетический алгоритм, найти заданную строку длины $N = 100$, состоящую из символов конечного алфавита размерности $K = 33$. Фитнесс-функция F сообщает о количестве позиций в оцениваемом текущем решении, значения в которых не совпадают со значениями в целевой (эталонной) строке. Таким образом, минимальное значение известно и достигается при $F = 0$. Заметим, что при таком выборе фитнесс-функции описание алгоритма не зависит от значения параметра N .

Исходя из особенностей класса задачи, при разработке алгоритма особое значение придавалось эффективному использованию операций скрещивания с целью накопления и сохранения «правильных» генов (символов, совпадающих со значениями в целевой строке) [7]. В итоге удалось разработать многопопуляционный генетический алгоритм, при реализации которого возможно достичь результата в 98 % от оптимума (получать хромосому-решение, содержащую 98 % «правильных» генов), используя только операции скрещивания, без использования операций мутации.

© И.О. Лукьянов, Ф.А. Литвиненко, 2020

Предложенный алгоритм условно состоит из трёх последовательных этапов:

На *начальном этапе* случайным образом генерируются начальные популяции (множество хромосом-решений заданной длины N) и некоторым образом (используя операции скрещивания) формируется элитные части этих популяций. Цель данного этапа – сохранение генного материала каждой популяции, т. е. концентрация как можно большего числа “правильных” генов в нескольких хромосомах-решениях.

На *обменном этапе* популяции обмениваются лучшими хромосомами-решениями таким образом, чтобы с одной стороны максимально приблизиться к оптимальному решению, а с другой – не допустить вырождения популяций.

На *заключительном этапе*, когда операции скрещивания и обмена практически исчерпывают себя, дальнейший процесс оптимизации осуществляется с использованием операций мутации. С их помощью алгоритм восполняет недостающий полезный генетический материал, который был утерян в процессе работы алгоритма или изначально отсутствовал в начальных популяциях.

В данной работе приведены результаты компьютерных экспериментов, в которых вышеописанный многопопуляционный генетический алгоритм использовался при нахождении оптимальных решений для следующих тестовых фитнес-функций:

$$F_1 = \sum_1^N x_i^2 \text{ – фитнес-функция с равнозначными факторами,}$$

$$F_2 = \sum_1^L x_i^2 + \sum_{L+1}^N \alpha x_i^2 \text{ – фитнес-функция со значимыми и малозначимыми факторами,}$$

$$F_3 = \sum_1^N x_i^2 + \beta \xi + C \text{ – фитнес-функция со случайным влиянием.}$$

Для корректности проведения экспериментов и наглядности сравнения их результатов с оригинальными [4–6] параметры данной задачи были приближены к параметрам предыдущей. Для этого все $N = 100$ факторов каждой фитнес-функции, принимающих значения в диапазоне $x_i \in [-15; 15]$, были разбиты на 30 интервалов (квантов). Так же для фитнес-функции F_2 определены параметры: L ($1 \leq L \leq N$) – количество значимых факторов и α ($0 < \alpha < 1$) – коэффициент при малозначимых факторах. Таким образом, F_2 будет содержать L значимых и $N - L$ малозначимых факторов. Для имитации случайного влияния в фитнес-функции F_3 введена случайная величина ξ , коэффициент β и аддитивная константа $C = 30$. В экспериментах с новыми тестовыми фитнес-функциями алгоритм модифицирован естественным образом, а точнее операция мутации: теперь она производит смещение k ($1 \leq k \leq N$) случайных генов хромосомы-решения на случайное количество квантов в случайном направлении.

Значительным отличием новых тестовых фитнес-функций от предыдущей является то, что хромосома-решение с меньшим количеством оптимальных «правильных» генов в некоторых случаях может давать результат лучше, чем хромосома-решение с большим количеством «правильных» генов. С предыдущей функцией такого произойти не могло, поскольку каждый ген улучшал значение F только тогда, когда его значение было «правильным». Например, если все значения генов хромосомы-решения для новых фитнес-функций будут равняться 1 или -1 , то ни один из этих генов не будет «правильным», однако значение F будет ближе к минимальному ($F = 0$), чем в случае, когда 99 генов будут «правильными», а единственный ген – равняться 15 или -15 .

По этой причине, при сравнительно таких же параметрах запуска, данные функции будут не в состоянии достичь 98 % от оптимума с использованием исключительно операций скрещивания, так как некоторая часть «правильных» генов будет вытеснена вышеописанными хромосомами-решениями. Вследствие этого факта, операция мутации с некоторой вероятностью должна применяться как на *обменном*, так и на *заключительном* этапах алгоритма, а не только на *заключительном*, как это было ранее. Кроме того, при проведении экспериментов было условлено, что алгоритм будет работать до достижения точки ближайшей к оптимуму, в данном случае это $F_i = 1$ (в случае с F_3 это $1 + C$).

Эксперименты с фитнес-функцией F_2 сначала проводились с коэффициентом при малозначимых факторах $\alpha = 0.1$. Их количество постепенно увеличивалось от 0 до 80 с шагом в 10 – ось абсцисс на рис. 1. Соответственно количество значимых факторов L постепенно уменьшалось от 100 до 20 с таким же шагом. В последующих экспериментах коэффициент α был уменьшен до 0.01.

В экспериментах с фитнес-функцией F_3 сравнивались результаты по мере изменения коэффициента β при стандартной нормально распределенной случайной величине $\xi = N(0,1)$.

Результаты экспериментов

Эксперименты с тестовой функцией F_1 показали, что параметры операции мутации оказывают большое влияние на скорость схождения к оптимуму (см. таблицу).

Стоит отметить, что обмен большим количеством хромосом-решений, описанный в [5], не влиял на результаты экспериментов, в которых общее количество итераций было большим. Однако, при использовании

наиболее эффективных параметров, приведенных в таблице: $cross_prob = 0.9$, $mute_prob = 0.1$, $k = 1$ – изменение частоты обмена лучшими пятью хромосомами-решениями на каждые 10 итераций, привело к уменьшению общего количества итераций до 280. Кроме этого, уменьшение начального этапа описанного в [5] до 15 итераций вместо 25 улучшило результаты, и общее количество итераций в среднем составило 275.

Результаты экспериментов с тестовой функцией F_2 , показаны на рис. 1, демонстрируют зависимость количества итераций от количества малозначимых факторов и их коэффициента значимости. На рис. 1 приставка *все* – означает количество итераций, которые понадобились для нахождения оптимума с использованием как, значимых так и малозначимых факторов. В свою очередь, приставка *знач* – означает количество итераций, которые понадобились для нахождения оптимума с использованием только значимых факторов. Наличие малозначимых факторов увеличивает общее количество итераций, в некоторых случаях – при $L = 60$ и $\alpha = 0.01$ – почти в 2 раза. Такой результат вполне ожидаем. Его можно объяснить тем, что большое количество операций мутации и скрещивания проводится с малозначимыми факторами, что не так существенно влияет на значение фитнес-функции, и соответственно, замедляет нахождение оптимума. Стоит отметить, что уменьшение значения α увеличило количество итераций нужных для поиска решения. Это объясняется тем, что операции скрещивания и мутации с малозначимыми факторами при меньшем значении коэффициента α имеют еще меньшее влияние на результат фитнес-функции.

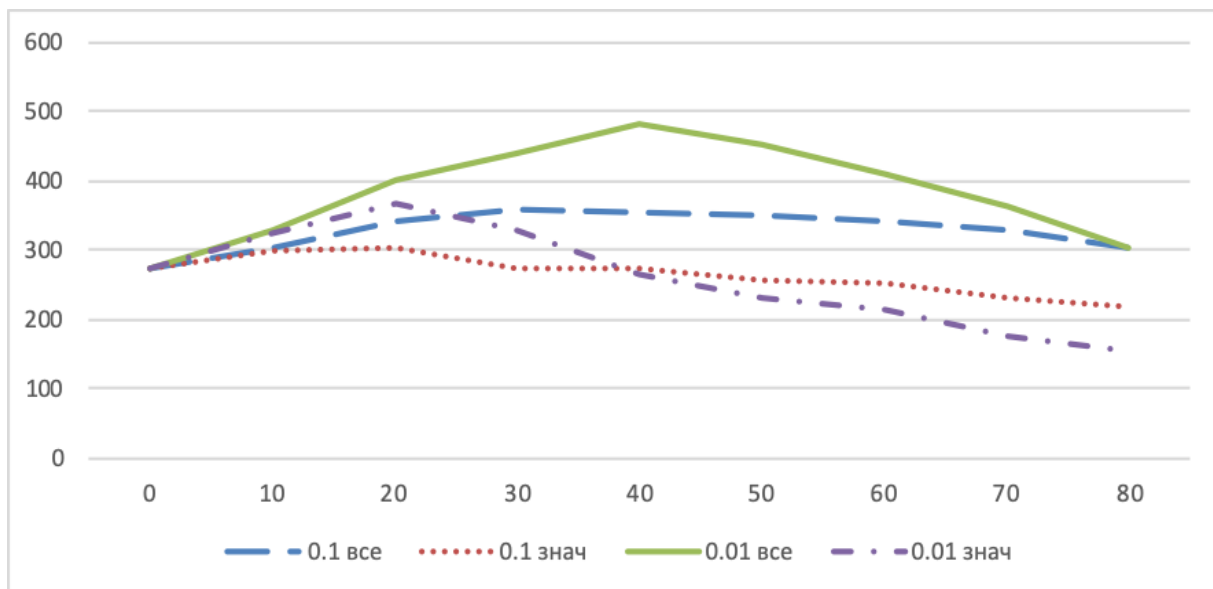


Рис. 1. Количество итераций для нахождения оптимальных значений значимых и всех факторов при разных коэффициентах и количестве малозначимых факторов

Таблица. Среднее количество итераций при изменении параметров мутации

$cross_prob$	$mute_prob$	k	Среднее количество итераций
0.9	0.1	5	833
0.85	0.1	5	1024
0.95	0.1	5	776
0.95	0.05	5	839
0.95	0.15	5	1029
0.95	0.1	3	451
0.95	0.1	1	290
0.9	0.1	1	285

При небольшом количестве малозначимых факторов (10, 20, 30) также увеличивается число итераций необходимых для поиска оптимальных значений только значимых факторов. Так как значимых факторов все еще достаточно много, нахождение оптимального значения для каждого из них является достаточно объемной задачей, а некоторые операции мутации и скрещивания в процессе экспериментов будут влиять только на малозначимые факторы. При этом для нахождения оптимальных значений факторов, при большом количестве малозначимых факторов, понадобилось меньше итераций, чем для фитнесс-функции F_1 . Можно предположить, что причиной этому является значительное уменьшение размерности задачи, так как количество значимых параметров гораздо меньше 100.

Стоит отметить, что для нахождения оптимальных значений только значимых факторов, влияние коэффициента α работает неоднозначно – меньшее его значение уменьшает количество итераций при малом количестве значимых факторов, но увеличивает при большом количестве значимых факторов.

Для сравнения на рис. 2 показано количество итераций, которые потребовались для нахождения оптимального решения при уменьшении общего количества факторов до значения L – количества значимых факторов (т.е. при полном исключении из рассмотрения малозначимых факторов).

Таким образом, наличие малозначимых факторов негативно сказывается на скорости схождения многопопуляционного параллельного генетического алгоритма. Соответственно их выявление и последующее исключение из компьютерной модели является важной задачей для улучшения скорости работы алгоритма.

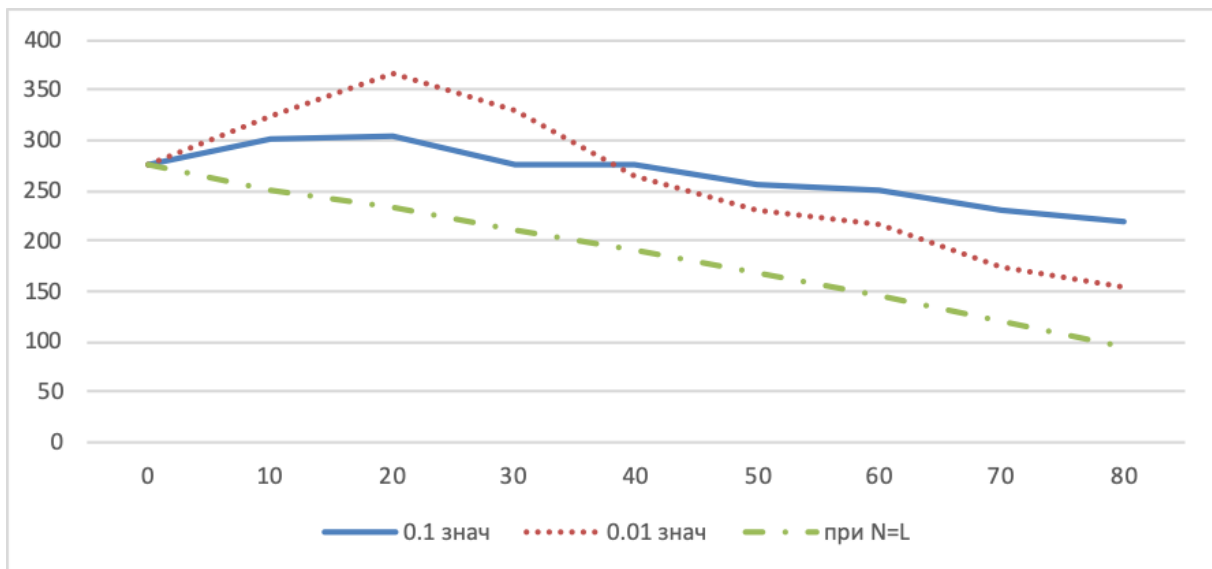


Рис. 2. Количество итераций для нахождения оптимальных значений значимых факторов при разных коэффициентах и количестве малозначимых факторов

На рис. 3 показано результаты экспериментов с тестовой функцией F_3 (сплошная линия). График демонстрирует зависимость количества итераций от значения β . Из-за присутствия в фитнесс-функции F_3 стандартной нормально распределенной случайной величины ξ , при значении $\beta = 4.5$, и при всех значимых параметрах ($L = 0$), среднее количество итераций выросло с 275 до 313. При $\beta = 9$, среднее количество итераций выросло до 359. Данный результат является вполне ожидаемым – большее влияние случайного воздействия приводит к увеличению количества итераций необходимых для поиска оптимального значения. Причиной такого эффекта является то, что для каждой хромосомы-решения значение фитнесс-функции F_3 искажено случайным влиянием, что негативно сказывается на процессе селекции генетического алгоритма.

При этом, дальнейшее увеличение β приводит к уменьшению количества итераций. Это можно объяснить тем, что влияние случайного воздействия было слишком большим, и алгоритм заканчивал работу, допуская нахождение ближайшего к оптимуму результата. Поскольку при приближении к оптимуму скорость схождения значительно замедляется [4], это позволяет пропустить значительный этап работы, в некоторых случаях 30 – 40 % итераций. Пунктиром на рис. 3 обозначено количество итераций необходимых алгоритму для нахождения хромосомы-решения, значение F_1 для которой, будет равно значению фитнесс-функции F_3 без влияния случайного воздействия. Например: при значении $\beta = 18$, алгоритму потребовалось 330 итераций для нахождения оптимальной хромосомы-решения для функции $F_3 = 31$, хотя ее же значение с функцией F_1 (т. е. без учета случайного влияния) равняется 85 – C . Изначально, работая с функцией F_1 , алгоритм достиг бы такого же значения фитнесс-функции за 185 итераций.

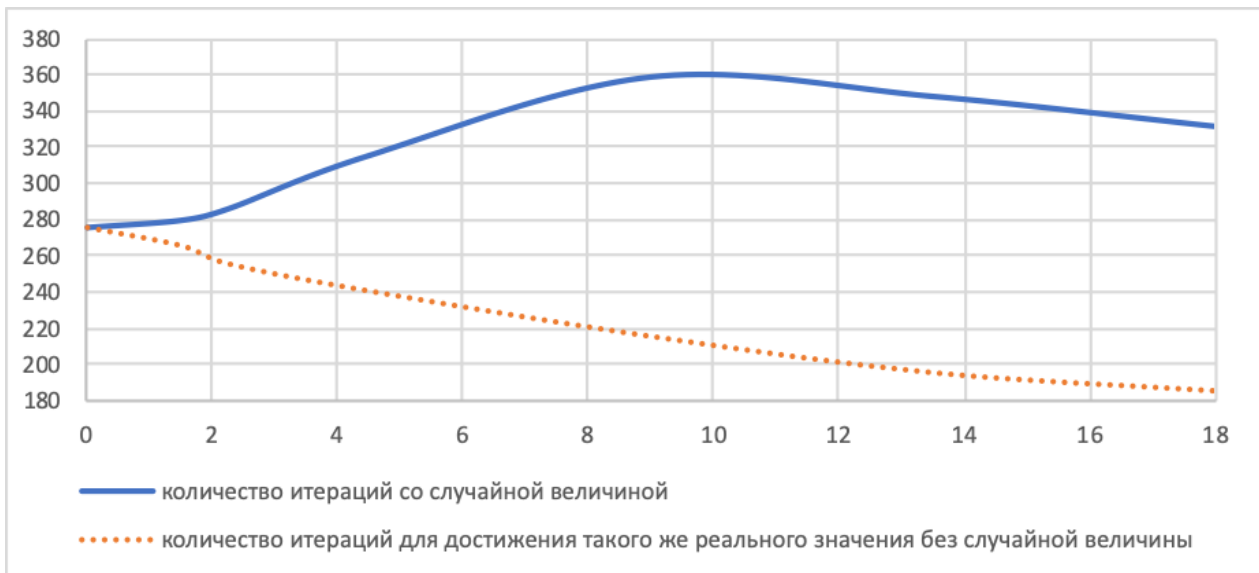


Рис. 3. Зависимость количества итераций от значения β

Учитывая это обстоятельство, были проведены эксперименты с динамическим значением β . Для этого на каждом процессоре, то есть для каждой популяции, запоминалось минимальное значение фитнес-функции, и β вычислялось, как определенный процент от этого значения.

Как показано на рис. 4, при динамическом значении β отсутствует «переломная» точка и количество итераций постепенно увеличивается. В сравнении с предыдущим случаем (при постоянном значении β) разница заключается в том, что алгоритм проходит все этапы (значение фитнес-функции F_1 для всех хромосом-решений находится ближе к оптимальному значению), что и обуславливает увеличение количества итераций.

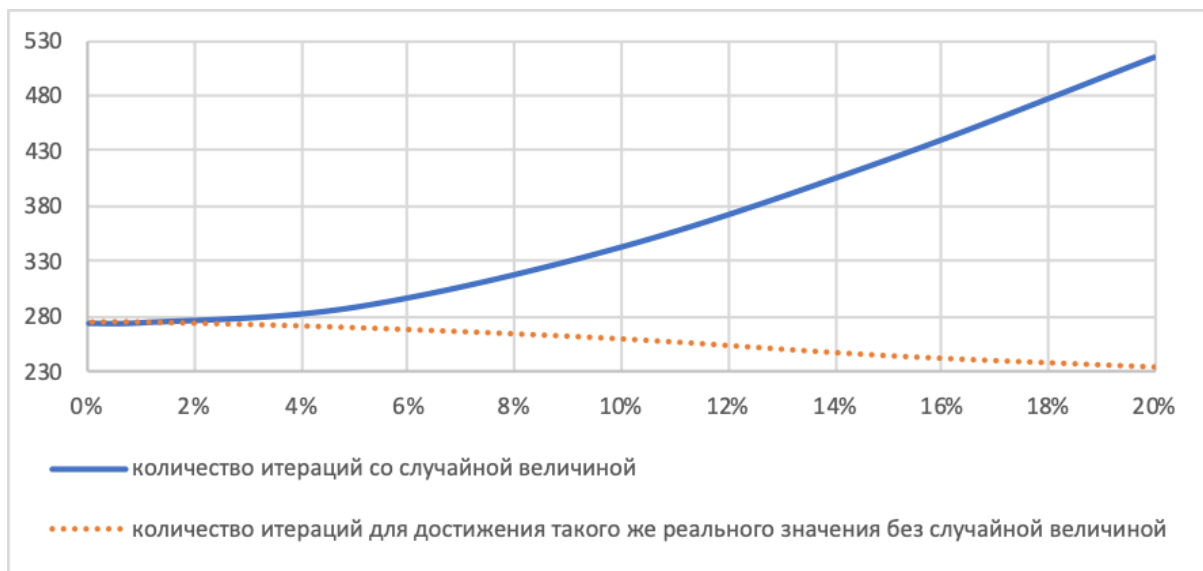


Рис. 4. Зависимость количества итераций от значения β заданного динамически

Выводы

Из результатов проведенных экспериментов видно, что скорость сходимости и эффективность генетического алгоритма сильно зависят от выбранных параметров запуска. При их выборе также стоит учитывать и особенности фитнес-функции, которые часто не известны заранее, и их нужно оценивать в ходе самих экспериментов. В результате многопопуляционный генетический алгоритм, разработанный для конкретного класса задач, успешно удалось адаптировать под особенности других фитнес-функций. Присутствие малозначимых факторов значительно замедляет скорость нахождения оптимального решения, причем этот эффект в значительной мере зависит от их количества и степени значимости. Так же стоит отметить, что для нахождения оптимальных значений исключительно значимых факторов, привносящих

наибольший вклад в результат прогона модели, влияние коэффициента α работает неоднозначно – меньшее его значение уменьшает количество итераций при малом количестве значимых факторов, но так же увеличивает его при большом количестве значимых факторов. Случайное влияние, которое присутствует в модели, негативно сказывается на работе алгоритма. В зависимости от природы этого случайного влияния, оно может сказываться либо в количестве итераций – значительно замедляя скорость нахождения оптимальной хромосомы-решения, либо в существенном отклонении результата – найденная хромосома-решение, на самом деле, будет далеко не оптимальным результатом. Таким образом, важной задачей компьютерного моделирования является нахождение малозначимых факторов и определение природы случайного влияния. В дальнейших работах планируется проведение экспериментов с присутствием этих двух факторов в одной фитнес-функции и изучение их влияния на результаты работы алгоритма.

Литература

1. Пепеляев В.А. Об эволюционных подходах к оптимизации имитационного моделирования. *Компьютерная математика*. 2005. № 1. С. 48–54.
2. Пепеляев В.А. О планировании оптимизационно-имитационных экспериментов. *Кибернетика и системный анализ*. 2006. № 6. С. 112–125.
3. Пепеляев В.А., Чёрный Ю.М. О возможностях применения генетических алгоритмов в оптимизационно-имитационных экспериментах. *Теория оптимальных решений*. 2019. С. 100–109.
4. Лукьянов И.О., Литвиненко Ф.А., Криковлюк Е.А. Особенности реализации параллельной версии многопопуляционного генетического алгоритма. *Компьютерная математика*. 2018. № 2. С. 21–29.
5. Лукьянов И.О., Литвиненко Ф.А., Криковлюк Е.А. О повышении эффективности параллельной версии многопопуляционного генетического алгоритма. *Теория оптимальных решений*. 2019. № 18. С. 116–122.
6. Литвиненко Ф.А., Лукьянов И.О., Криковлюк Е.А. Использование разнообразности начальной популяции в многопопуляционном генетическом алгоритме. *Компьютерная математика*. 2019. № 1. С. 116–123.
7. Лукьянов И.О., Литвиненко Ф.А., Коваль В.П. О выборе размера начальной популяции для параллельной версии многопопуляционного генетического алгоритма. Сборник материалов конференции IntSol-2019 «Теория принятия решений». 2019. С. 95-96.

References

1. Pepelyaev V. On evolutionary approaches to the optimization of simulation modeling. *Computer Mathematics*. 2005. N 1. P. 48–54.
2. Pepelyaev V. On the planning of optimization-simulation experiments. *Cybernetics and system analysis*. 2006. N 6. P. 112–125.
3. Pepelyaev V., Cherny Y. On the possibilities of using genetic algorithms in optimization and simulation experiments. *Theory of Optimal Solutions*. 2019. P. 100–109.
4. Lukianov I., Lytvynenko F., Krykovliuk O. Features of the implementation of a parallel version of the multipopulation genetic algorithm. *Computer math*. 2018. N 2. P. 21–29.
5. Lukianov I., Lytvynenko F., Krykovliuk O. On increasing the efficiency of a parallel version of a multipopulation genetic algorithm. *Theory of optimal solutions*. 2019. N 18. P. 116–122.
6. Lytvynenko F., Lukianov I., Krykovliuk O. Using the diversity of the initial population in a multipopulation genetic algorithm. *Computer Mathematics*. 2019. N 1. P. 116–123.
7. Lukianov I., Lytvynenko F., Koval V. On the choice of the size of the initial population for the parallel version of the multipopulation genetic algorithm. Collection of materials of the IntSol-2019 conference “Decision Making Theory”. 2019. P. 95–96.

Получено 10.03.2020

Об авторах:

Лукьянов Игорь Олегович,
аспирант.

Количество научных публикаций в украинских изданиях – 9.

Литвиненко Федор Антонович,
аспирант.

Количество научных публикаций в украинских изданиях – 9.

Место работы авторов:

Институт кибернетики имени В.М. Глушкова НАН Украины,
03187, м. Київ-187,
проспект Академика Глушкова, 40.
Тел.: +38 (044) 526-11-96.
E-mail: ihorlukianov@gmail.com,
fedirlytvynenko@gmail.com