Секция 2
МЕТОДЫ И ТЕХНОЛОГИИ МАТЕМАТИЧЕСКОГО И ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ СИСТЕМ

УДК 681.3.05

АЛГОРИТМ ПОИСКА ЭКСТРЕМУМА МУЛЬТИМОДАЛЬНЫХ ЦЕЛЕВЫХ ФУНКЦИЙ

канд. техн. наук, доц. А. Ф. ОСЬКИН
(Полоцкий государственный университет, Беларусь);

Д. А. ОСЬКИН
(Белорусский государственный экономический университет, Минск)

Аннотация. *Предлагается новый алгоритм численного поиска экстремума мультимодальных целевых функций. Он относится к многочисленной группе алгоритмов искусственного интеллекта, объединенных общим названием – «Алгоритмы роевого интеллекта». Приводится описание реализации алгоритма в виде консольного приложения на языке программирования С++, а также результаты численных экспериментов по поиску с помощью разработанного приложения экстремумов тестовых функций из библиотеки тестовых функций для оптимизации.*

Методы роевого интеллекта – группа итеративных методов, моделирующих поведение организованной колонии насекомых. В ряде работ [1–3] показано, что с помощью этих методов возможно нахождение решения, близкого к оптимальному за приемлемое время. Этим и объясняется популярность этих методов при решении NP – сложных задач, для которых точные алгоритмы поиска решения известны, но неприемлемы из-за своих временных характеристик. Мы проанализировали три роевых алгоритма – метод роя частиц, метод пчелиного роя и метод светлячков – и разработали простой алгоритм, который при проведении численных экспериментов показал вполне приемлемые результаты.

Алгоритм состоит из следующих шагов.

Шаг 1. Генерация роя частиц, случайным образом рассеянных по области поиска решения.

Шаг 2. Определение «лучшей» частицы, т.е. частицы, для которой значение целевой функции наиболее близко к экстремуму.

Шаг 3. Для каждой из частиц роя определение расстояния между нею и «лучшей» частицей.

Шаг 4. Для каждой из частиц роя определение направления смещения – по кратчайшему пути в сторону «лучшей» частицы.

Шаг 5. Для каждой из частиц роя расчёт нового положения в соответствии с формулой

$$\boldsymbol{X}_i(t+1) = \boldsymbol{X}_i(t) + k \cdot \boldsymbol{\Delta X}_i,$$

где $\boldsymbol{X}_i(t+1)$ – координаты i-й частицы на t+1 шаге итерационного процесса;

$\boldsymbol{X}_i(t)$ – координаты i-й частицы на предыдущем шаге итерационного процесса;

k – эмпирический коэффициент;

$\boldsymbol{\Delta X}_i$ – смещение i-й частицы по всем координатным осям.

Шаг 6. Определение новой «лучшей» частицы.

Шаг7. Если не выполнено запланированное число итераций – переход на шаг 3. Иначе -переход на шаг 8.

Шаг 8. Вывод координат «лучшей» частицы.

Шаг 9. Конец.

Для проверки работоспособности и эффективности алгоритма, нами написано консольное приложение, реализующее описанный алгоритм. Использовался язык С++. Некоторые результаты численных экспериментов с приложением приводятся ниже. Мы пользовались библиотекой тестовых функций, размещённой по адресу http://www.sfu.ca/~ssurjano. Для решения задачи оптимизации использовался рой из 20 частиц. Значение эмпирического коэффициента во всех расчетах, результаты которых здесь приводятся, принималось равным 1,5.

Результаты моделирования. Целевая функция – функция Растригина

$$f(x1, x2) = 20 + [x1^2 - 10 \cdot \cos(2 \cdot \pi \cdot x1)] + [x2^2 - 10 \cdot \cos(2 \cdot \pi \cdot x2)].$$

Это мультимодальная с несколькими локальными минимумами. Глобальный минимум достигается в точке [0, 0].

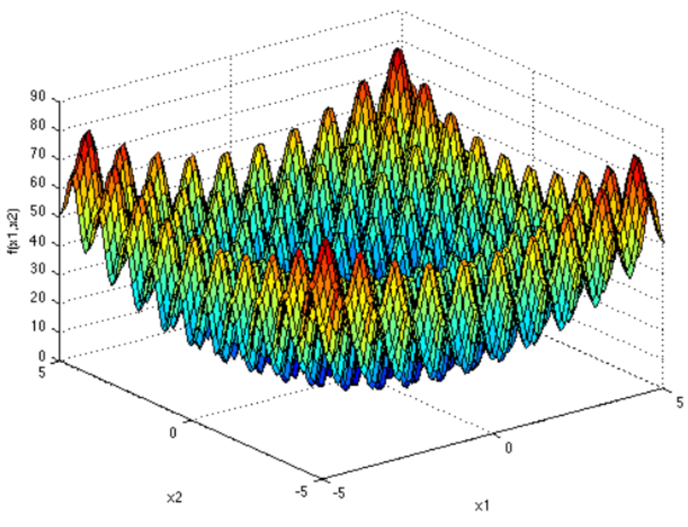

Рисунок 1. – Функция Растригина

Таблица 1. – Изменение положения частиц в процессе поиска экстремума для функции Растригина

	Первые десять частиц роя										
X1	4,00	-1,00	1,00	1,00	-3,00	5,00	-3,00	-2,00	0,00	2,00	Исходное расположение частиц
X2	-5,00	1,00	-1,00	-4,00	4,00	5,00	-5,00	0,00	3,00	5,00	
X1	-2,00	0,50	-0,50	-0,50	1,50	-2,50	1,50	1,00	0,00	-1,00	Первая итерация
X2	1,00	-2,00	-1,00	0,50	-3,50	-4,00	1,00	-1,50	-3,00	-4,00	
X1	1,00	-0,25	0,25	0,25	-0,75	1,25	-0,75	-0,50	0,00	0,50	Третья итерация
X2	-2,00	-0,50	-1,00	-1,75	0,25	0,50	-2,00	-0,75	-1,03	0,50	
X1	0,25	-0,06	0,06	0,06	-0,19	0,31	-0,19	-0,13	0,00	0,13	Пятая итерация
X2	-0,50	-0,13	-0,25	-0,44	0,06	0,13	-0,50	-0,19	0,00	0,13	

Целевая функция –функция Била

$$f(x1,x2)=(1{,}5-x1+x1\cdot x2)^2+(2{,}25-x1+x1\cdot x2^2)^2+(2{,}625-x1+x1\cdot x2^3)^2.$$

Функция Била – мультимодальная, с резкими подъёмами по углам области определнния. Глобальный минимум достигается в точке [3, 0,5] и равен нулю.

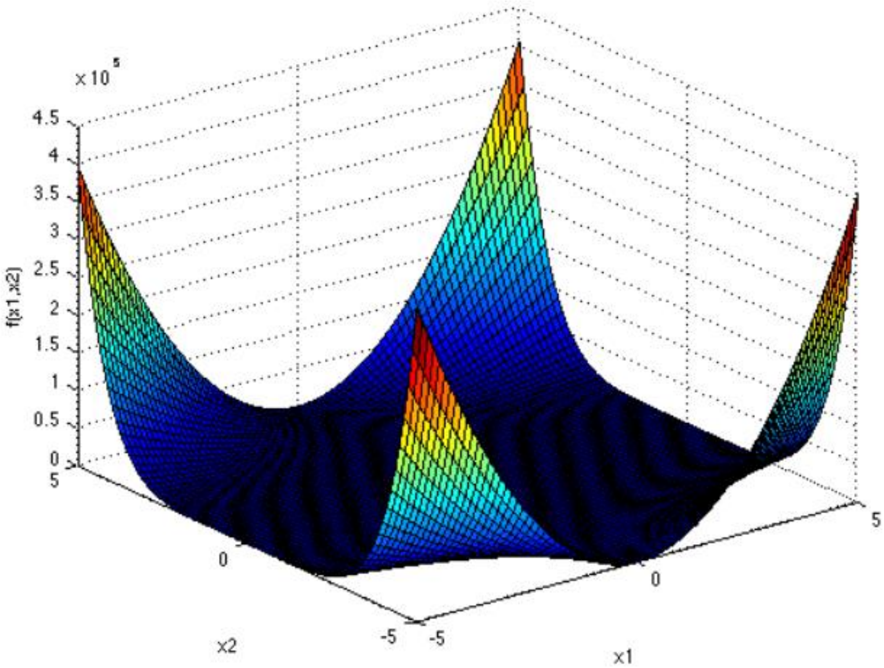

Рисунок 2. – Функция Била

Таблица 2. – Изменение положения частиц в процессе поиска экстремума для функция Била.

	Первые десять частиц роя										
X1	0,00	-2,00	3,00	-3,00	2,00	-2,00	1,00	3,00	4,00	3,00	Исходное расположение частиц
X2	1,00	-1,00	1,00	-3,00	1,00	3,00	-1,00	4,00	0,00	1,00	
X1	1,50	2,50	0,00	3,00	0,50	2,50	1,00	0,00	-0,50	0,00	Первая итерация
X2	-2,00	-1,00	-2,00	0,00	-2,00	-3,00	-1,00	-3,50	-1,50	-2,00	
X1	3,00	3,25	2,63	3,38	2,75	3,25	2,88	2,63	2,50	2,63	Третья итерация
X2	0,25	0,50	0,25	0,75	0,25	0,00	0,50	-0,13	0,38	0,25	
X1	2,91	2,97	2,81	3,00	2,84	2,97	2,88	2,81	2,78	2,81	Пятая итерация
X2	0,44	0,50	0,44	0,56	0,44	0,38	0,50	0,34	0,47	0,44	

Выводы

1. Несмотря на простоту, алгоритм оказался достаточно эффективен и позволяет получать вполне приемлемые результаты.
2. К недостаткам алгоритма следует отнести редкие, но тем не менее имеющие место случаи попадания частиц в локальные экстремумы.
3. Для устранения этого недостатка, по аналогии с методом роя частиц, мы предполагаем ввести дополнительные рандомизированные эмпирические коэффициенты.

ЛИТЕРАТУРА

1. «Particle Swarm Optimization». Proceedings of IEEE International Conference on Neural Networks IV: 1942-1948.
2. Kennedy, J.; Eberhart, R.C. Swarm Intelligence — Morgan Kaufmann, 2001. — ISBN 1-55860-595-9.
3. Poli, R. An analysis of publications on particle swarm optimisation applications (англ.) // Technical Report CSM-469 : journal. — Department of Computer Science, University of Essex, UK, 2007.