Перевод статьи: Minaeva Yu.V.
Adaptive modification of the particle swarm method based on dynamic correction of the trajectory of movement of individuals in the population.
Business Informatics. 2016. No. 4 (38). P. 52–59.

Адаптивная модификация метода роя частиц на основе динамической коррекции траектории движения особей в популяции

Ю.В. Минаева

старший преподаватель кафедры систем автоматизированного проектирования и информационных систем
Воронежский государственный технический университет
Адрес: 394026, г. Воронеж, Московский пр-т, д. 14
E-mail: julia_min@mail.ru

Аннотация

Методы эволюционного поиска успешно применяются для решения разнообразных задач оптимизации и моделирования ввиду своей универсальности и относительной простоты практической реализации. Однако большой проблемой при их использовании является преждевременная сходимость вычислительного алгоритма вследствие неполного исследования пространства поиска. Это происходит в том случае, когда все частицы попадают в область первого обнаруженного, возможно, локального оптимума, и не могут из нее выбраться. Для решения этой проблемы необходима разработка управляющих процедур, корректирующих перемещение особей в популяции.

В статье предлагается адаптивная модификация метода роя частиц, позволяющая осуществлять динамическое изменение траектории движения частиц для нахождения наиболее перспективных локаций. В основе метода лежит возможность изменения вектора перемещения индивидуально для каждой частицы, в зависимости от результативности выполнения предыдущей итерации. Для этого в предлагаемую модификацию канонического метода добавлены процедуры выбора направления и динамического изменения свободных параметров движения частицы. В отличие от канонической версии роевого алгоритма, в котором все особи популяции стремятся приблизиться к одной частице с наилучшим найденным значением, в новой модификации каждая частица самостоятельно выбирает направление движения и может изменить его в случае, если оно будет признано неэффективным. Такой подход позволяет снизить вероятность преждевременной сходимости алгоритма и лучше исследовать заданную область поиска, что особенно важно для многоэкстремальных функций со сложным рельефом. Предложенный метод был проверен на стандартном наборе тестовых функций непрерывной оптимизации и показал высокую эффективность при относительно небольших затратах времени и вычислительных ресурсов.

Ключевые слова: оптимизация, эволюционные алгоритмы, метод роя частиц, преждевременная сходимость, адаптация алгоритма, гибридизация алгоритмов.

Цитирование: Minaeva Yu.V. Adaptive modification of the particle swarm method based on dynamic correction of the trajectory of movement of individuals in the population // Business Informatics. 2016. No. 4 (38). P. 52–59. DOI: 10.17323/1998-0663.2016.4.52.59.

Введение

При проектировании сложных технических и экономических систем часто возникает задача оптимального выбора внутренних характеристик системы, описывающих ее структуру или поведение. К таким задачам относятся формирование производственной программы предприятия, выбор оборудования и технологии производства, обоснование компоновочных схем, выбор и оценка рисков инвестиционных проектов и др. Для увеличения эффективности и скорости реализации процедур оптимального поиска разработано множество методов, однако практически все из них имеют ограничения, связанные с характером математической модели исследуемой системы. Наиболее универсальными с этой точки зрения являются алгоритмы управляемого перебора, основанные на процессах эволюционного развития биологических популяций. Одним из таких методов является метод роя частиц (МРЧ), использующий модели поведения сложных самоорганизующихся систем с социальной структурой. Такие системы состоят из простых взаимодействующих агентов, каждый из которых ведет себя независимо от других, но в результате поведение всей многоагентной системы оказывается разумным [1].

Потенциальные решения в МРЧ представляются в виде популяции живых организмов, каждый из которых занимает определенную позицию внутри роя. Смыслом существования всех организмов популяции является повышение степени индивидуальной полезности за счет перемещения в локации с лучшими значениями целевой функции. Для этого частицы постоянно обновляют свои координаты, используя как собственные знания, так и опыт, накопленный остальными организмами популяции [2].

К настоящему времени разработано множество модификаций канонического МРЧ, однако многие из них сохраняют недостатки, свойственные первоначальному алгоритму. Одним из наиболее перспективных направлений исследований в области эволюционных алгоритмов является изучение адаптационных свойств МРЧ, улучшение которых позволит повысить эффективность и универсальность поисковых процедур.

Целью данной статьи является разработка адаптивной модификации МРЧ, позволяющей производить динамическую коррекцию траектории движения частиц для более эффективного исследования заданной области поиска. В основе предлагаемого метода лежит возможность выбора каждой частицей направления, при движении в котором возрастает ее полезность для популяции.

1. Классический метод роя частиц

Пусть рой состоит из n частиц. Каждая из частиц роя в любой момент может быть описана своими координатами $x_i = \{x_{i1}, x_{i2}, \ldots, x_{id}\}$ и скоростью $v_i = \{v_{i1}, v_{i2}, \ldots, v_{id}\}$, где i – номер частицы ($i = 1, \ldots, n$), d – размерность пространства поиска. Тогда весь рой частиц в k-й момент времени характеризуется вектором координат $x_k = \{x_1, x_2, \ldots, x_n\}$ и вектором скоростей всех частиц $v_k = \{v_1, v_2, \ldots, v_n\}$. Согласно каноническому методу роя частиц, разработанному Кеннеди и Эберхардом [1], итерации выполняются по следующей схеме:

$$v_{k+1} = \alpha v_k + \beta r_1 (p_k - x_k) + \gamma r_2 (g_k - x_k),$$
$$x_{k+1} = x_k + v_{k+1},$$

где p_k и g_k – координаты наилучшего решения, найденного самой частицей и всем роем, соответственно, α, β, γ – свободные параметры алгоритма, r_1 и r_2 – случайные числа в интервале [0, 1]. Коэффициенты α, β и γ определяют степень влияния каждой из трех составляющих на скорость частицы. Величина α отвечает за инерционность движения частицы. Если значение α близко к 1, то частица продолжает свой путь, исследуя таким образом все пространство поиска. В противном случае частица стремится к наилучшему значению (своему или социальному) и будет находиться в области вокруг него. Значение β отражает влияние когнитивного компонента, т.е. стремление частицы вернуться к самому «хорошему» с точки зрения целевой функции значению, найденному ею ранее. Величина γ выражает социальную составляющую скорости, т.е. тенденцию перемещения частицы к текущему лучшему решению, найденному остальными частицами [2].

Недостатками канонического метода являются [2]:

✦ возможность выхода координат частицы за пределы области допустимых значений функции;

✦ преждевременная сходимость алгоритма к первому экстремуму (в общем случае локальному) и невозможность дальнейшего поиска.

2. Модификации метода роя частиц

Для устранения недостатков метода разработано множество модификаций, одни из которых направлены на улучшение работы всего алгоритма в целом, а другие предназначены для решения конкретного класса задач.

Все разработанные модификации можно отнести к одной из следующих групп:

✧ модификация когнитивной составляющей;

✧ модификация социальной составляющей;

✧ подбор свободных параметров алгоритма;

✧ гибридизация алгоритмов.

К наиболее существенным модификациям когнитивной составляющей канонического алгоритма относятся учет в формуле скорости не только «положительного» опыта частицы, но и отрицательного, т.е. стремление удалиться от «плохих» значений целевой функции [3], а также возможность принудительного перемещения частицы при длительной стагнации ее координат [4].

Модификации социальной составляющей предполагают учет влияния не только наилучшего на данный момент решения, но и текущих значений остальных частиц. К этой группе можно отнести такие алгоритмы, как МРЧ с полной информацией, в котором наибольшее влияние на движение частицы имеют частицы с более «хорошими» значениями [5], и МРЧ, основанный на отношении «значение–расстояние», где степень влияния каждой частицы зависит от близости ее расположения [2].

Влияние социальной составляющей на эффективность процедур оптимального поиска в большой степени определяется топологической структурой популяции, т.к. именно от этой характеристики зависит размер подмножества частиц, с которыми может обмениваться своим опытом каждая отдельная частица [2]. Исследования эффективности и сходимости МРЧ и его модификаций при различных топологических структурах показывают, что топологии со слабой связностью частиц, т.е. с небольшим количеством соседей, позволяют эффективнее исследовать пространство поиска и уменьшают вероятность преждевременной сходимости алгоритма [2, 5, 6].

Эффективность и надежность МРЧ во многом зависит от соблюдения правильного баланса между стадиями исследования пространства поиска и локализацией экстремума. Для регулирования соотношения между этими стадиями используются такие свободные параметры алгоритма, как α, β и γ, для которых в разных работах предлагают использовать как константные значения, так и зависящие от времени. Например, для коэффициента α разработаны следующие схемы приращений значений коэффициента [7, 8]:

- линейная

$$\alpha_k = \alpha_{max} - \frac{(\alpha_{max} - \alpha_{min})}{T_{max}} k,$$

где α_{max} и α_{min} – допустимые максимальное и минимальное значения коэффициента;

T_{max} – максимально возможное число итераций;

- нелинейная

$$\alpha_k = \alpha_{min} + \frac{(\alpha_{max} - \alpha_{min})}{T_{max}^2} \left(T_{max} - k\right)^2,$$

- экспоненциальная

$$\alpha_k = \alpha_{min} + (1 - \alpha_{min}) e^{-\lambda k},$$

где λ – заданная константа.

При использовании линейной и нелинейной схем обязательным является предварительное задание T_{max}.

Для коэффициентов β и γ также рекомендуется использовать как константные значения [1, 2, 9], так и зависимые от времени [9, 10].

Подобная привязка изменения коэффициентов ко времени выполнения алгоритма может привести к недостаточно эффективному поиску решения, поскольку невозможно точно предугадать, на какой именно итерации оптимум будет обнаружен и локализован. Для устранения этого недостатка в работах [8, 9] предлагаются адаптивные модификации метода роя частиц, позволяющие более объективно управлять процессом оптимизации.

В работе [9] предлагается разделить процесс решения на четыре стадии, в зависимости от величины разброса частиц: исследование области поиска, локализация оптимума, стагнация, выход из состояния стагнации. Каждой стадии соответствует определенная стратегия изменения коэффициентов. На этапе исследования области поиска коэффициент β увеличивается, а γ уменьшается, при локализации оптимума β и γ изменяются незначительно, в случае стагнации коэффициенты незначительно увеличиваются, при выходе из состояния стагнации β уменьшается, а γ увеличивается. Коэффициент инерционности авторы предлагают изменять в зависимости от величины изменения значения целевой функции.

Метод роя частиц, как и другие эволюционные алгоритмы, легко может быть использован в составе гибридных схем. Например, в работах [11–13] для определения новой скорости частицы предлагается использовать операции селекции, скрещивания и мутации, взятые из генетического алгоритма. Кроме того, для повышения качества метода применяются локальный поиск и дифференциальная эволюция [14]. Метод роя частиц также может использоваться в сочетании с неэволюционными алгоритмами или их составными частями (например, в работе [14] показано использование операций растяжения, отражения и перемещения из метода деформированного многоугольника Нелдера–Мида [15]). Частным случаем гибридизации являются и многороевые алгоритмы [16], состоящие из нескольких роев, каждый из которых, в общем случае, обладает своим набором параметров.

3. Метод роя частиц с адаптацией траектории движения

Канонический метод роя частиц предполагает стремление всех частиц к одному центру с наилучшим на данный момент значением целевой функции. Однако такой процесс может привести, во-первых, к преждевременной стагнации алгоритма, а во-вторых, к уходу частиц из перспективных локаций и потере значений функции в них.

Для устранения указанных недостатков метода предлагается проводить динамическую коррекцию траектории движения каждой частицы за счет адаптивного выбора направления и изменения степеней влияния собственного и социального опыта. С этой целью в классическую версию метода добавляются следующие процедуры:

✦ процедура выбора каждой частицей x_i своего «социального лидера» x_j, $j = 1, ..., n$, с помощью метода турнирной селекции, заимствованного из генетического алгоритма;

✦ процедура коррекции значений коэффициентов α, β, γ.

Алгоритм динамической адаптации метода роя частиц включает следующие шаги:

1. Инициализация значений скоростей, координат частиц и свободных коэффициентов

$$v_i^0 = rnd(D_v), x_i^0 = rnd(D_x),$$

$$p_i^0 = x_i^0, y_i = i,$$

$$\alpha_i^1 = \alpha\_max, \beta_i^1 = \beta\_max, \gamma_i^1 = \gamma\_min, ,$$

где D_v, D_x – области допустимых значений скоростей и координат частиц;

y – массив, содержащий номера частиц, к которым направлена социальная составляющая вектора скорости частиц.

Устанавливается номер итерации $k = 1$.

2. Вычисление значений целевой функции $f(x_i^{k-1})$ и обновление списка наилучших найденных значений для всех частиц

$$p_i^k = \begin{cases} x_i^{k-1} - when\ f(x_i^{k-1}) < f(p_i^{k-1}). \\ otherwise,\ p_i^{k-1} \end{cases}$$

3. Если $k < 2$, то переход к п. 4, иначе выполняется процедура динамической адаптации параметров движения частицы, включающая два этапа:

а) выбор направления движения частицы в зависимости от изменения ее полезности по методу турнирной селекции:

$$y_i = \begin{cases} arg\ min\left(f(\tilde{X}_i)\right) - when\ f(x_i^k) > f(x_i^{k-1}) + \delta f, \\ y_i, - otherwise \end{cases}$$

где $\tilde{X}_i$ – случайное подмножество частиц, выбираемое для частицы x_i, $\tilde{X}_i \subseteq X$;

б) корректировка коэффициентов, отвечающих за инерционность движения и степень влияние когнитивной и социальной составляющих скорости:

$$\alpha_i^{k+1} = \alpha_i^k - \delta\varkappa,$$
$$\beta_i^{k+1} = \beta_i^k - \delta\varkappa,$$
$$\gamma_i^{k+1} = \gamma_i^k + \delta\varkappa,$$

где $\delta\varkappa$ – поправка к значениям коэффициентов α, β, γ, определяемая следующим образом:

$$\delta\varkappa = \frac{f(x_i^{k-1}) - f(x_i^k)}{f(x_i^{k-1})},$$

Для корректной работы алгоритма после пересчета коэффициентов выполняется проверка на принадлежность новых значений допустимым интервалам:

$$z_j = \begin{cases} z\_min_j, & when\ \ z_j < z\_min_j, \\ z_j, & when\ \ z\_min_j \le z_j \le z\_max_j, \\ z\_max_j, & when\ \ z_j > z\_max_j, \end{cases}$$

где $z = \{\beta_i^k, \gamma_i^k, \alpha_i^k\}$ – массив текущих значений коэффициентов;

$z\_min = \{\alpha\_min, \beta\_min, \gamma\_min\}$,

$z\_max = \{\alpha\_max, \beta\_max, \gamma\_max\}$ – массивы с заданными минимальными и максимальными значениями для каждого коэффициента, $j = 1, ..., 3$.

4. Пересчет скорости и координат частиц:

$$v_i^k = \alpha_i^k v_i^{k-1} + \beta_i^k r_1 (p_i^{k-1} - x_i^{k-1}) + \gamma_i^k r_2 (x_{y_i}^{k-1} - x_i^{k-1}),$$
$$x_i^k = x_i^{k-1} + v_i^k$$

5. Если выполнено условие останова алгоритма $\left|f(x_i^k) - f(x_i^{k-1})\right| \le \varepsilon$, где ε – допустимая погрешность вычислений, то происходит окончание работы алгоритма, иначе устанавливается номер итерации $k = k + 1$ и осуществляется переход к п. 2.

Для оценки эффективности предлагаемого алгоритма сравним его работу с каноническим МРЧ (PSO), МРЧ с полной информацией (FIPS) [5] и адаптивным МРЧ (APSO), предложенным в [9]. Разработанную модификацию с динамической коррекцией траектории движения частиц обозначим как TPSO. Экспериментальное исследование методов проводилось при следующих параметрах:

Таблица 1.

Набор тестовых функций для проверки алгоритма

Название	Формула	Интервал
Сферическая функция	$F_1(x)=\sum_{i=1}^{n}x_i^2$	$x\in(-5,\,5)$
Функция Розенброка	$F_2(x)=\sum_{i=1}^{n-1}\left(100\left(x_{i+1}-x_i^2\right)^2+\left(x_i-1\right)\right)$	$x\in(-2.5,2.5)$
Функция Де Йонга 2	$F_3(x)=\dfrac{-100}{100\cdot\left(x_1^2-x_2\right)+\left(1-x_1\right)^2+1}$	$x\in(-5,\,5)$
Функция Растригина	$F_4(x)=10n+\sum_{i=1}^{n}\left(x_i^2-10cos\left(2\pi x_i\right)\right)$	$x\in(-5,\,5)$

✧ для всех методов устанавливается размер популяции 30 частиц и ограничение по времени 2000 итераций;

✧ для PSO коэффициент α уменьшается по линейной схеме от $\alpha_{max}=0.9$ до $\alpha_{min}=0.4$, коэффициенты $\beta=\gamma=1.49618$ [2];

✧ для FIPS $\alpha=0.7298$, $\beta=\gamma=2.05$[5];

✧ для методов APSO и TPSO устанавливаются границы $\alpha_{max}=0.9$, $\alpha_{min}=0.4$, $\beta_{max}=2.5$, $\beta_{min}=1.5$, $\gamma_{max}=2.5$, $\gamma_{min}=1.5$ [9].

Используемые тестовые функции (*таблица 1*) взяты из рекомендуемого стандартного набора тестовых задач непрерывной оптимизации [2] и позволяют проверить качество поиска экстремумов для функций с различным рельефом пространства поиска. Две тестовые функции (сферическая и Розенброка) являются унимодальными, остальные – сложными мультимодальными. Для всех функций размерность координатного пространства $n=10$.

Для сравнения различных вариантов МРЧ используются такие показатели, как эффективность поисковых процедур, количество итераций, затраченных на поиск оптимума и время решения. В качестве времени решения берется разница между системным временем компьютера, измеренным до начала вычислений и после их окончания. В *таблице 2* приведены усредненные значения данных показателей. Наилучшее значение в каждой строке выделяется полужирным начертанием.

По результатам вычислительного эксперимента можно сделать вывод, что на использованном наборе тестовых функций предлагаемый алгоритм показал высокую эффективность как для унимодальных, так и для сложных мультимодальных задач. При этом для поиска экстремума ему понадобилось меньше итераций и меньше затрат времени работы процессора, чем остальным модификациям МРЧ (на трех из четырех тестовых функциях). По своим показателям предложенный метод близок к APSO, однако его преимуществом является более простая практическая реализация и меньшее время, затрачиваемое на поиск решения.

Таблица 2.

Показатели эффективности различных вариантов МРЧ

Функция	Показатели	Методы решения			
		PSO	FIPS	APSO	TPSO
$F_1(x)$	Эффективность	**100%**	**100%**	**100%**	**100%**
	Число итераций	523,4	492,1	478,5	**475,4**
	Время решения	0,311	**0,302**	0,324	0,305
$F_2(x)$	Эффективность	**100%**	**100%**	**100%**	**100%**
	Число итераций	591,2	524,7	**489,1**	498,3
	Время решения	0,408	0,397	0,415	**0,380**
$F_3(x)$	Эффективность	90,1%	96,5%	**100%**	**100%**
	Число итераций	754,5	685,4	621,2	**613,8**
	Время решения	0,634	0,605	0,615	**0,594**
$F_4(x)$	Эффективность	92,5%	97,1%	**100%**	**100%**
	Число итераций	712,9	647,5	597,2	**584,1**
	Время решения	0,612	0,594	0,618	**0,590**
Среднее значение	Эффективность	95,7%	98,4%	**100%**	**100%**
	Число итераций	645,4	587,4	546,5	**542,9**
	Время решения	0,491	0,474	0,493	**0,467**

Заключение

В данной статье предложена гибридная модификация МРЧ, позволяющая производить адаптивную коррекцию траектории движения частицы, в зависимости от эффективности выполнения процедуры оптимального поиска на предыдущей итерации. Подобная схема решения позволяет частицам быстро перемещаться в наиболее перспективные локации за счет динамического изменения степеней влияния когнитивной и социальной составляющих их скорости. Рассмотренная модификация канонического метода апробирована на стандартных тестовых задачах непрерывной оптимизации. По результатам тестирования можно сделать вывод, что применение динамической коррекции траектории движения частиц позволяет повысить эффективность процесса поиска глобального оптимума и уменьшить время эволюции по сравнению с существующими модификациями МРЧ. ■

Литература

1. Kennedy J., Eberhart R.C. Particle swarm optimization // Proceedings of the IEEE International Conference on Neural Networks (ICNN‘95), 27 November – 01 December 1995, Perth, Australia. Vol. 4. P. 1942–1948.
2. Карпенко А.П., Селиверстов А.П. Глобальная безусловная оптимизация роем частиц на графических процессорах архитектуры CUDA // Наука и образование: Электронное научно-техническое издание. 2010. №4. [Электронный ресурс]: http://technomag.edu.ru/doc/142202.html (дата обращения: 01.07.2016).
3. Yang C., Simon D. A new particle swarm optimization technique // Proceedings of the 18th International Conference on Systems Engineering (ICSEng‘05), 16–18 August 2005, Las Vegas, USA. P. 164–169.
4. Xie X., Zhang W., Yang Z. Adaptive particle swarm optimization on individual level // Proceedings of the 6th International Conference on Signal Processing (ICSP’02), 26–30 August 2002, Beijing, China. P. 1215–1218.
5. Mendes R., Kennedy J., Neves J. The fully informed particle swarm: Simpler, maybe better // IEEE Transactions on Evolutionary Computation. 2004. Vol. 8. No. 3. P. 204–210.
6. Kennedy J., Mendes R. Population structure and particle swarm performance // Proceedings of the 2002 Congress on Evolutionary Computation (CEC’02), 12–17 May 2002, Washington, USA. P. 1671–1676.
7. Parsopoulos K.E., Vrahatis M.N. (Eds.) Particle swarm optimization and intelligence: Advances and applications. N.Y.: IGI Global, 2010.
8. Clerc M. Particle swarm optimization. London: ISTE, 2006.
9. Zhan Z., Zhang J., Li Y., Chung H.S.H. Adaptive particle swarm optimization // IEEE Transactions on Systems, Man, and Cybernetics. Part B. 2009. Vol. 39. No. 6. P. 1362–1381.
10. Ratnaweera A., Halgamuge S.K., Watson H.C. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients // IEEE Transactions on Evolutionary Computation. 2004. Vol. 8. No. 3. P. 240–255.
11. Angeline P.J. Using selection to improve particle swarm optimization // Proceedings of the 1998 IEEE International Conference on Evolutionary Computation (ICEC’98). 4–9 May 1998, Anchorage, Alaska, USA. P. 84–89.
12. Chen Y.P., Peng W.C., Jian M.C. Particle swarm optimization with recombination and dynamic linkage discovery // IEEE Transactions on Systems, Man, and Cybernetics. Part B. 2007. Vol. 37. No. 6. P. 1460–1470.
13. Andrews P.S. An investigation into mutation operators for particle swarm optimization // Proceedings of the 2006 IEEE Congress on Evolutionary Computation, 16–21 July 2006, Vancouver, BC, Canada. P. 1044–1051.
14. Liang J.J., Suganthan P.N. Dynamic multi-swarm particle swarm optimizer with local search // Proceedings of the 2005 IEEE Congress on Evolutionary Computation (CEC’05), 2–5 September 2005, Edinburgh, Scotland. P. 522–528.
15. Gimmler J., Stützle T., Exner T.E. Hybrid particle swarm optimization: An examination of the influence of iterative improvement algorithms on performance // Proceedings of the 5th International Workshop “Ant Colony Optimization and Swarm Intelligence” (ANTS 2006), 4-7 September 2006, Brussels, Belgium. P. 436–443.
16. Jordan J., Helwig S., Wanka R. Social interaction in particle swarm optimization, the ranked FIPS, and adaptive multi-swarms // Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation (GECCO’08), 12–16 July 2008, Atlanta, USA. P. 49–56.