Паттерны популяционных алгоритмов непрерывной глобальной оптимизации

А. П. Карпенко
Московский государственный технический университет имени Н.Э. Баумана
Москва, Россия
apkarpenko@mail.ru

***Аннотация*. Рассматриваем популяционные алгоритмы (П-алгоритмы) непрерывной глобальной оптимизации. Выделяем такие сущности, как свободные параметры, пространство соседства, особь, популяция, объединение особей, окрестность особи, а также следующие эволюционные операторы: инициализация популяции, окончание поиска, кодирование особей, рандомизация, селекция, скрещивание, управление популяцией, локальный поиск. На основе предложенных способов описания указанных сущностей представляем паттерны наиболее известных П-алгоритмов: эволюционные алгоритмы, поведенческие алгоритмы, вдохновленные живой и неживой природой, а также человеческим обществом.**

Ключевые слова: глобальная оптимизация; популяционный алгоритм; метаэвристический алгоритм; паттерн популяционного алгоритма

I. Введение

П-алгоритмы (population-based algorithms) непрерывной глобальной поисковой оптимизации в разных публикациях называют поведенческими, интеллектуальными, мета-эвристическими, вдохновленными (инспирированными) природой, роевыми, многоагентными и т.д. [1]. П-алгоритмы многочисленны и весьма разнообразны, так [2] представлено более 130 таких алгоритмов, и продолжают появляться новые алгоритмы. Известно небольшое число работ, в той или иной мере посвященных различным аспектам анализа П-алгоритмов. Так, в фундаментальных работах [3, 4] с использованием псевдо-кода представлены алгоритмы и операторы эволюционных алгоритмов. В работе [5] общие схемы П-алгоритмов и их операторов представлены также в форме псевдокода. Высокоуровневый формализм, восходящий к работам Дж. Холланда и ориентированный на описание адаптивных П-алгоритмов, предложен в работе [6] в контексте обсуждения подходов, которые могут использоваться для разработки новых П-алгоритмов.

II. Постановка задачи

Рассматриваем детерминированную задачу непрерывной глобальной безусловной минимизации

$$\min_{X \in R^{|X|}} f(X) = f(X^*) = f^*,$$

где $|X|$ - размерность вектора варьируемых параметров X; $f(X)$ - целевая функция (Ц-функция) со значениями в пространстве R^1; X^*, f^* - искомые оптимальные решение и значение Ц-функции соответственно. Поиск решения поставленной задачи начинается в $|X|$-мерном параллелепипеде $\Pi = \{X \mid X^- \le X \le X^+\}$, где X^-, X^+ - его нижняя и верхняя границы. Рассматриваем также аналогичную задачу глобальной условной минимизации. Полагаем, что фитнесс-функция $\varphi(X)$ П-алгоритма также подлежит минимизации.

III. Сущности П-алгоритмов

Различаем следующие сущности: детерминированные (deterministic) и стохастические (stochastic); статические (static), динамические программно изменяемые (program) и динамические адаптивные (adaptive). Таким образом, определение сущности содержит описатели $system \in \{det, stoch\}$, $control \in \{stat, prog, adapt\}$. Исключая эволюционные операторы, выделяем такие сущности, как свободные параметры и мета-параметры, пространство соседства, особь, популяция, объединение особей, окрестность особи, след особи и др.

Мета-параметры – это вспомогательные свободные параметры, которые определяют значения основных свободных параметров. Описание мета-параметров П-алгоритма имеет вид

parameters $\overline{P}\,|_{ID}^{system.control}$ = <мета-параметры>:< ограничения>.

Пространство соседства особей определяет запись

space $NAME(A, P)\,|_{ID}^{system.control}$: <метрика пространства>.

Здесь $NAME \in \{R_\varphi, R_X, R_T\}$; R_φ, R_X, R_T - фитнесс-пространство, пространство поиска и топологическое пространство соответственно; A, P – списки аргументов и свободных параметров.

Объединение особей в пространстве R_α, $\alpha \in \{\varphi, X, T\}$ определяет выражение вида union R_α $NAME(A, P)\,|_{ID}^{systemcontrol}$= <элементы>.

Окрестность особи $s_i \in S$ в том же пространстве R_α представляет собой часть этого пространства, элементы которой в смысле метрики пространства R_α близки к положению особи s_i :

area $NAME(s_i, A, P)\Big|_{R_\alpha .ID}^{system\,control}$: <элементы>.

След особи s_i популяции S на интервале времени $[0:t]$ представляет собой совокупность пар $(X_i(\tau), \varphi_i(\tau))$ для всех $\tau \in [0:t]$:

track $Tr(s_i, t, A, P)\,|^{det.stat}\,\{(X_i^0, \varphi_i^0),...,(X_i^t, \varphi_i^t)\}$.

Основными являются следующие эволюционные операторы: инициализация популяции, окончание поиска, кодирование особей, рандомизация, селекция, скрещивание, управление популяцией, локальный поиск.

Инициализация сущности Ξ :

$$\text{init } \Xi\,(A, P)\Big|_{ID}^{systemcontrol} = \Xi(0) = \Xi^0 .$$

Рандомизация поиска. Оператор рандомизации эволюции особи $s_i \in S$ определяет выражение

$$\text{rand}\,(s_i, A, P)\,|_{ID}^{systemcontrol} = X_i' .$$

Рандомизирующий оператор скрещивания особей $\{s_{i_1}, s_{i_2},..., s_{i_n}\} \in S$ в общем случае имеет вид

$$\text{rand.crossing}\,(A, P)\Big|_{n\times m.ID}^{systemcontrol} = \{s'_{j_1}, ..., s'_{j_m}\}, \quad \{s_{i_1},..., s_{i_n}\} \in A .$$

Управление популяцией. Из числа операторов управления популяцией определены операторы сжатия, расширения, репликации расщепления диссимиляции популяции и др. Оператор сжатия популяции S, например, имеет вид

$$\text{compression } (S, A, P)\Big|_{ID}^{systemcontrol} = S' .$$

IV. Паттерны П-алгоритмов

Представляем в качестве примера четыре следующих П-алгоритма.

A. *Дифференциальная эволюция DE*

Рассматриваем *DE*-алгоритм, в котором базовую особь [1] выбирают из текущей популяции равномерно случайно.

1) Инициализация

Инициализация пользователем детерминированных статических свободных параметров $|S|, |X|, X^-, X^+, a, \xi_b$ (оператор init P$\Big|_{DM}^{det.stat}$), где *DM* - Decision-Maker. Равномерно случайно распределяем особей популяции в области Π (оператор init S$\Big|_{uniform}^{stoch.stat}$).

2) Эволюция популяции

Для каждого $i \in [1:|S|]$ используем рандомизирующий оператор скрещивания rand.cross $\Big|_{3\times1.DE}^{stoch.stat}$:

$$x'_{i,j} = \begin{cases} x_{i_1,j} + a\,(x_{i_2,j} - x_{i_3,j})\,|\,\xi_b, & j = j_1 = U_1[1:|X|], \\ x_{i,j}, & \text{иначе}, \end{cases}$$

$$\{s_i, s_{i_1}, s_{i_2}, s_{i_3}\} \in S\ ;\ i = U_1[1:|S|],\ i_k = U_1[1:|S|],\ k \in [1:3],$$
$$i_1 \ne i_2 \ne i_3 \ne i,\ j = [1:|X|] .$$

Производим селекцию особей (оператор select$|_{\varphi\,best}^{det.stat}$):

$$s_i(t+1) = \begin{cases} s'_i, & \varphi(s'_i) < \varphi(s_i), \\ s_i(t+1), & \text{иначе}. \end{cases}$$

Здесь $U_1[1:|S|]$ - целое число, равномерно случайно распределенное в интервале $[1:|S|]$.

3) Завершение эволюционного процесса (end$|^{any}$)

Описатель *any* означает, что условия окончания поиска не фиксированы.

Паттерн DE-алгоритма

specifications

parameters $P\,|_{DM}^{det.sat} = (|S|, |X|, X^-, X^+, a, b)$):

$a \in [0; 2]$, $\xi_b \in [0;1]$.

individual $S : X = (x_j,\ j \in [1:|X|]) \in \Pi \subset R^{|X|}$.

population $S = \{s_i,\ i \in [1:|S|]\}$.

initialization

init P$(P)\Big|_{DM}^{det.stat} = P$.

init S$\,(S, P)\Big|_{uniform}^{stoch.stat} = S(0) = S^0$.

evolution

rand.cross $(S, P)\,|_{3\times1.DE}^{stoch.stat}$, $i \in [1:|S|]$.

select $(S, S')\,|_{\varphi.best}^{det.stat} = S(t+1)$.

termination

end $(S, P)\,|^{any} = (\tilde{X}^*, \tilde{f}^*)$.

B. *Алгоритм оптимизации роем частиц PSO.*

1) Инициализация

Инициализация пользователем свободных параметров (оператор $\text{init P}\Big|_{DM}^{det.stat}$). Инициализация популяции путем равномерно случайного распределения особей популяции в области Π, а векторов их начальных скоростей $\Delta X_i,\ i\in[1:|S|]$ в области Π_Δ (оператор $\text{init S}\Big|_{uniform}^{stoch.stat}$).

2) Эволюция популяции

К каждой из особей $s_i \in S$ применяем рандомизирующий оператор скрещивания $\text{rand.crossing}\Big|_{|S|\times 1.PSO}^{stoch.stat}$:

$$X_i' = X_i + b_I \Delta X_i^- + U_{|X|}(0;b_C)\otimes(X_i^* - X_i) + $$
$$+U_{|X|}(0;b_S)\otimes(X_i^{**} - X_i),$$

$$\varphi(X_i^*) = \min_{\tau\in[0:t]} \varphi(X_i(\tau)),\quad \varphi(X_i^{**}) = \min_{j\in N_i} \varphi(X_j^*).$$

Здесь $U_{|X|}(0;b)$ - $|X|$-мерный вектор случайных чисел равномерно распределенных в интервале $[0;b]$; N_i - множество соседей особи s_i [1].

3) Завершение эволюционного процесса ($\text{end}\,|^{any}$)

Паттерн алгоритма *PSO*

specifications

parameters $P\,|_{DM}^{det.stat} = (|S|, |X|, X^-, X^+, b_I, b_C, b_S)$:

$b_I = 0{,}7298$, $b_C = b_S = 1{,}49618$.

individual S: $X = (x_j,\ j\in[1:|X|]) \in \Pi \subset R^{|X|}$.

population $S = \{s_i,\ i\in[1:|S|]\}$.

space $R_T\ |^{det.stat}$: $\mu_{R_T}(s_i, s_j) = r(s_i, s_j)$, $i, j\in[1:|S|]$.

track $Tr(s_i, t, P)\,|^{det.stat} = \{X_i^0, ..., X_i^t\}$, $i\in[1:|S|]$.

initialization

$\text{init P}(P)\Big|_{DM}^{det.stat} = P$.

$\text{init S}(P)\Big|_{uniform}^{stochstat} = S(0) = S^0$.

evolution

$\text{select}(Tr_i, P)\,|_{best}^{det.stat} = X_i^*$, $i\in[1:|S|]$.

$\text{select}(G_i, P)\,|_{best}^{det.stat} = X_i^{**}$, $i\in[1:|S|]$.

$\text{rand.crossing}\ (s_i, X_i^*, X_i^{**}, P)\Big|_{|S|\times 1.PSO}^{stoch.stat} = s_i(t+1)$; $i\in[1:|S|]$.

termination

$\text{end}\ (P)\,|^{any} = (\tilde{X}^*, \tilde{f}^*)$.

Использованы следующие обозначения: μ_{R_T} - метрика близости особей в пространстве R_T; $r(s_i, s_j)$ - расстояние между вершинами s_i, s_j в графе соседства $G = G_T(S)$.

C. Электромагнитный алгоритм EM

1) Инициализация

Инициализация пользователем свободных параметров (оператор $\text{init P}\Big|_{DM}^{det.stat}$). Программная инициализация популяции путем равномерно случайного распределения особей в области Π (оператор $\text{init } S\Big|_{uniform}^{stoch.stat}$).

2) Эволюция популяции.

К каждой из особей $s_i \in S,\ i \ne i_b$ применяем рандомизирующий оператор скрещивания $\text{rand.crossing}\Big|_{|S|\times 1.EM}^{stoch.stat}$:

$$X_i' = X_i + \lambda_1 U_1(1)\frac{F_i}{\|F_i\|_E}\otimes V_i,\quad X_{i_b}' = X_{i_b};$$

$$F_i = \sum_{j=1,\ j\ne i}^{|S|} F_{i,j} = \sum_{j=1,\ j\ne i}^{|S|}\begin{cases}(X_j - X_i)\dfrac{q_i\ q_j}{\|X_j - X_i\|_E^2}, & \varphi_j < \varphi_i,\\ (X_i - X_j)\dfrac{q_i\ q_j}{\|X_j - X_i\|_E^2}, & \varphi_i \le \varphi_j;\end{cases}$$

$$q_i = \exp\left(-|X|\ \frac{\varphi_i - \varphi_{i_b}}{\sum\limits_{j\in[1:|S|],\ j\ne i}(\varphi_j - \varphi_{i_b})}\right);$$

$$v_{i,j} = \begin{cases}(x^+ - x_{i,j}), & F_{i,j} > 0,\\ (x_{i,j} - x^-), & F_{i,j} \le 0,\end{cases}\quad j\in[1:|X|],\ j\ne i\ .$$

Для каждой из особей $s_i \in S$ реализуем линейный стохастический локальный поиск алгоритмом RLS (оператор $\text{local}\,|_{RLS}^{stoch.stat}$):

$$X_i'' = X_i' + \lambda_2 V_i,\quad i\in[1:|S|],$$

$$v_{i,j} = u_{sign}^{\pm 1} U_1(0;1),\ j\in[1:|X|],\ \lambda_2 = \alpha \max_{j\in[1:|X|]}(x^+ - x^-).$$

3) Завершение эволюционного процесса ($\text{end}\,|_{\hat{t}}^{det.stat}$).

Условие окончания поиска $t = \hat{t}$.

Паттерн EM-алгоритма

specifications

parameters $P\,|_{DM}^{det.stat} = (|S|, |X|, X^-, X^+, \lambda_1, \alpha, \hat{\tau}_l, \hat{t})$:

$\alpha\in(0;1)$, $\hat{\tau}_l >> |X|$, $\hat{t} = 25\,|X|$.

space $R_X\ |^{det.stat}$: $\mu(s_i, s_j) = \|X_i - X_j\|_E$.

individuals S: $X(s) = (x_j,\ j\in[1:|X|]) \in \Pi \subset R^{|X|}$.

population $S = \{s_i,\ i\in[1:|S|]\}$.

initialization

$\text{init P}\Big|_{DM}^{det.stat} = P$.

init S $(P)\,\Big|_{uniform}^{stoch.stat} = S(0)$.
evolution
rand.crossing $(s_i\,,\,P)\,\Big|_{|S|\times 1.EM}^{stoch.stat} = s_i'$; $i\in[1:|S|]$.
local $(s_i',\,P)\,|_{RLS}^{stoch.stat} = s_i'' = s_i(t+1)$; $i\in[1:|S|]$.
termination
end $(P)\,|_{\hat t}^{det.stat} = (\widetilde{X}^*,\ \widetilde{f}^*)$.

D. Простой алгоритм эволюции разума SMEC

1) Инициализация

Инициализация пользователем свободных параметров (оператор init $\mathrm{P}|_{DM}^{det.stat}$). Программное равномерно случайное распределение особей $s_i^b,\, i\in[1:|S^b|]$ лидирующего объединения S^b в области Π (оператор init $\mathbf{S}\Big|_{uniform}^{stoch.stat}$). Аналогичная инициализация особей $s_i^w,\, i\in[1:|S^w|]$ отстающего объединения S^w (оператор init $\mathrm{S}\Big|_{uniform}^{stoch.stat}$).

2) Эволюция популяции

Локальный поиск в окрестности особи s_i^b (оператор local $|_{Normal}^{stoch.stat}$):

$$X_{i,j}^b = X_{i,1}^b + N_{|X|}(0,\sigma),\quad X_{i,1}^b = X_i^b,\quad j\in[2:n_g];$$

$$X_i'^b = X_{i,j_b}^b,\ \min_{j\in[1:n_g]}\varphi(X_{i,j}^b) = \varphi(X_{i,j_b}^b);\ i\in[1:|S^b|].$$

Аналогичный поиск в окрестности особи s_i^w, $i\in[1:|S^w|]$ (оператор local $|_{Normal}^{stoch.stat}$):

Формирование новой популяции (оператор dissimilation$\Big|_{SMEC}^{det.stat}$):

если $\varphi(s_i^b) > \varphi(s_k^w),\ k\in[1:|S^w|]$, то $s_k^w \to S^b,\ s_i^b \to S^w$;
$i\in[1:|S^b|]$;

если $\varphi(s_k^w) > \varphi(s_i^b)$, $i\in[1:|S^b|]$, то delete s_k^w , init $s_k^w\ |_{uniform}^{stoch.stat}$;
$k\in[1:|S^w|]$.

3) Завершение эволюционного процесса (end$|_{\varphi\ stagnation}^{det.stat}$)

Условие окончания итераций – стагнация вычислительного процесса.

Паттерн алгоритма SMEC

specifications
parameters
$P\,|_{DM}^{det.stat} = (|S|,\,|X|,\,X^-,\,X^+,\ |S^b|,\,|S^w|,\,\sigma\,,\,\delta t,\,\delta\varphi\ )$:
$|S^b| + |S^w| = |S|,\ \sigma > 0$.
individuals S: $X(s) = (x_j,\ j\in[1:|X|])\in\Pi\subset R^{|X|}$.
union $S^b\,|^{stoch.dynamic} = \{s_i^b,\, i\in[1:|S^b|]\}$.
union $S^w\,|^{stoch.dynamic} = \{s_i^w,\, i\in[1:|S^w|]\}$.
population $S = \{s_i,\, i\in[1:|S|]\} = S^b\cup S^w$.
initialization
init $\mathrm{P}\Big|_{DM}^{det.stat} = P$.
init union $S^b(P)\,\Big|_{uniform}^{stoch.stat} = S^b(0)$.
init union $S^w(P)\,\Big|_{uniform}^{stoch.stat} = S^w(0)$.
evolution
local $(s_i,\,P)\,|_{Normal}^{stoch.stat} = s_i'$, $i\in[1:|S|]$.
dissimilation $(\,S^b,\,S^w,\,P)\,\Big|_{SMEC}^{det.stat} = S' = S(t+1)$.
termination
end $(S,P)\,|_{\varphi\ stagnation}^{det.stat} = (\widetilde{X}^*,\ \widetilde{f}^*)$.

V. Заключение

С помощью предложенной нотации нами записаны паттерны основных П-алгоритмов, представленных в [1]. Этот опыт показал удобство и лаконизм предложенных средств. В развитии работы автор планирует разработку программной системы, предназначенной для автоматизированного синтеза П-алгоритмов.

Список литературы

[1] Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы вдохновленные природой. М.: Изд-во МГТУ им. Н.Э. Баумана, 2014. 446 с.

[2] Bo Xing, Wen-Jing Gao. Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms. Springer International Publishing Switzerland, 2014. 451 p.

[3] Evolutionary Computation. 1. Basic Algorithms and Operators / Edited by Thomas Back, David B. Fogel, Zbignew Michalevich. Institute of Physics Publishing, Bristol and Philadelphia, 2004. 339 p.

[4] Evolutionary Computation. 2. Advanced Algorithms and Operators / Edited by Thomas Back, David B. Fogel, Zbignew Michalevich. Institute of Physics Publishing, Bristol and Philadelphia, 2004. 339 p.

[5] Luke S. Essentials of Metaheuristics. Режим доступа: http://cs.gmu.edu/~sean/book/metaheuristics/ (дата обращения 06.01.2018).

[6] Brownlee J. Clever Algorithms: Nature-Inspired Programming Recipes. Режим доступа: http://www.cleveralgorithms.com/nature-inspired/index.html (дата обращения 06.01.2018).

[7] Скобцов Ю.А., Сперанский Д.В. Эволюционные вычисления: Учебное пособие. М.: Национальный Открытый Университет «ИНТУИТ», 2012. 331с.

[8] De Jong K.A. Evolutionary Computation: a Unified Approach. Massachusetts Institute of Technology, 2006. 256 p.