Особенности решения задачи о максимальном потоке методом дифференциальной эволюции

Ковалевич А.А.; Якимов А.И.
Кафедра АСУ, электротехнический факультет
ГУВПО «Белорусско-Российский университет»
г. Могилев, Беларусь
e-mail: ykm@tut.by

***Аннотация* — Рассматривается алгоритм дифференциальной эволюции и особенности его применения для решения задачи о максимальном потоке при наличии множества ограничений. Для улучшения работы алгоритма дифференциальной эволюции применен алгоритм адаптивных штрафов.**

Ключевые слова: дифференциальная эволюция, максимальный поток, ограничения, адаптация

I. Введение

Дифференциальная эволюция (ДЭ) – алгоритм многомерной математической оптимизации, относящийся к классу стохастических алгоритмов оптимизации (т. е. работает с использованием случайных чисел) и использующий некоторые идеи генетических алгоритмов.

Алгоритм ДЭ предназначен для нахождения глобального экстремума недифференцируемых, нелинейных, мультимодальных (имеющих, возможно, большое число локальных экстремумов) функций от многих переменных. Алгоритм был предложен Р. Сторном и К. Прайсом, впервые опубликован ими в 1995 году и разработан в дальнейшем в их более поздних работах.

В данной работе будет показано, как следует изменить алгоритм ДЭ, чтобы учитывать ограничения, налагаемые задачей о максимальном потоке.

II. Алгоритм дифференциальной эволюции

В работе [1] проведен сравнительный анализ стохастических алгоритмов оптимизации: роя частиц, имитации отжига и дифференциальной эволюции. Для проверки алгоритмов использовались функции *Branin RCOS*, модифицированная версия *RCOS*, *Easom* и *Goldstein-Price*. Показано, что алгоритм дифференциальной эволюции является лучшим алгоритмом, так как стабильно находит оптимум функции за минимальное время.

Подобно другим эволюционным алгоритмам ДЭ рассматривает случайную популяцию решений. Пусть начальная популяция решений состоит из *NP* индивидов и пространство поиска является *D*-мерным. Популяция *n*-ой итерации может быть представлена

$$\boldsymbol{X}(n) = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_{NP}),$$

где $\boldsymbol{x}$ – возможное решение в *D*-мерном пространстве поиска. Пусть существует некоторый критерий качества $f(\boldsymbol{x})$ такой, чтобы найти

$$\boldsymbol{x}^*: f(\boldsymbol{x}^*) = \min_{\boldsymbol{x}} f(\boldsymbol{x}).$$

Пусть $f: R^D \to R^+$ – целевая функция, которую требуется минимизировать. Алгоритм ДЭ включает три эволюционных процесса: мутацию, скрещивание, выбор.

Шаг 1. Оператор мутации случайным образом выбирает три различных индивида из текущей популяции и создает нового измененного индивида

$$\boldsymbol{v}_i = \boldsymbol{x}_{r1} + F \times (\boldsymbol{x}_{r2} - \boldsymbol{x}_{r3}) \mid F \in [0, 2],$$

где F – управляющий параметр, обычно выбираемый в интервале [0, 2], с наилучшими значениями в диапазоне [0,5, 0,9]. Оператор мутации является стохастическим отображением

$$T_m : S^{NP} \to S,$$

где $S = R^D$; S^{NP} – пространство популяции.

Шаг 2. Оператор скрещивания создает новое решение копированием компонентов мутационного вектора $\boldsymbol{v}_i$ и выбранного вектора $\boldsymbol{x}_i$

$$u_{ji} = \begin{cases} v_{ji} \mid r_b \le CR \lor j = r_r, \ i = 1, ..., NP; \\ x_{ji} \mid r_b > CR \lor j \ne r_r, \ j = 1, ..., D, \end{cases}$$

где $r_b = \text{rand}[0, 1]$ – случайное число в интервале [0, 1], $r_r =]\text{rand}[1, D][$ – случайное целое число в интервале [1, *D*], $CR \in [0, 1]$ – управляющий параметр скрещивания.

Оператор скрещивания является стохастическим отображением

$$T_r : S^2 \to S.$$

Шаг 3. Оператор выбора является детерминированным процессом в алгоритме ДЭ и выбирает индивида с лучшим значением целевой функции для следующего поколения

$$\boldsymbol{x}_i(n+1) = \begin{cases} \boldsymbol{u}_i \mid f(\boldsymbol{u}_i) \le f(\boldsymbol{x}_i); \\ \boldsymbol{x}_i \mid f(\boldsymbol{u}_i) > f(\boldsymbol{x}_i). \end{cases}$$

Процесс выбора описывается детерминированным оператором

$$T_s : S^2 \to S.$$

Оператор выбора гарантирует, что лучшее значение целевой функции не может быть пропущено, что приводит к быстрой сходимости. Алгоритм ДЭ может быть описан следующим образом

$$\boldsymbol{X}(n+1) = \{\boldsymbol{x}_i(n+1) \mid \boldsymbol{x}_i(n+1) = \\ = T_s \circ T_r \circ T_m(\boldsymbol{X}(n)), i = 1, ..., NP\}.$$

После этого шаги 1–3 повторяются, пока не будет сформировано нужное число поколений.

Алгоритм прост в реализации и использовании (содержит малое число управляющих параметров, требующих подбора: коэффициент мутации *F* и вероятность скрещивания *CR*), легко распараллеливается.

III. Постановка задачи о максимальном потоке

Математическая задача о максимальном потоке [2] в терминах линейного программирования формулируется следующим образом: найти неотрицательные значения $x_{ij}, \forall(v_i, v_j) \in E$, максимизирующие целевую функцию

$$F_{\max} = \sum_{j=2}^{n} x_{1j} \ \mid \ n = |V|;$$

или

$$F_{\max} = \sum_{i=1}^{n-1} x_{in} \ \mid \ n = |V|,$$

т. е.

$$\sum_{j=2}^{n} x_{1j} = \sum_{i=1}^{n-1} x_{in}, \qquad (1)$$

при ограничениях с учетом пропускной способности дуг:

$$0 \le x_{ij} \le c_{ij},\ i = 1, \ldots, n;\ j = 1, \ldots, n;\ i \ne j. \qquad (2)$$

и условиях сохранения потока в вершинах сети

$$\sum_{j=2}^{n} x_{kj} - \sum_{i=1}^{n-1} x_{ik} = 0;\ k = 2, \ldots, n-1; \qquad (3)$$

Условие (1) отражает величину максимального потока, который равен количеству вещества, вытекающего из источника, или количеству вещества, протекающего в сток. Условия (4.2) означают, что поток по каждой дуге должен быть неотрицательным и не превышать ее пропускной способности; из условия (4.3) следует, что количество вещества, притекающего в любую промежуточную вершину, равно количеству вещества, вытекающего из нее [3].

IV. Решение задачи о максимальном потоке методом дифференциальной эволюции

В настоящее время многие задачи по оптимизации требуют, чтобы были учтены ограничения, накладываемые на пространство поиска. Задача о максимальном потоке относится именно к этому классу задач.

Для добавления возможности учитывать ограничения в алгоритм ДЭ добавлен метод адаптивных штрафов (АШ) [4]. Суть метода АШ заключается в том, чтобы при вычислении целевой функции подсчитывать нарушение ограничений, умноженных на коэффициент штрафа. При этом, чем сложнее удовлетворить ограничению, тем больше коэффициент штрафа. Достоинством метода АШ следует считать то, что для алгоритма ДЭ не добавляется никаких параметров.

Для применения метода АШ вычисление целевой функции $f(\boldsymbol{x})$ должно быть изменено следующим образом:

$$F(X) = \begin{cases} f(\boldsymbol{x}) & \mid \boldsymbol{x} \in L; \\ \bar{f}(\boldsymbol{x}) + \sum_{j=1}^{m} k_j v_j(\boldsymbol{x}) & \mid \boldsymbol{x} \notin L. \end{cases}$$

где $F(X)$ – целевая функция, используемая в методе АШ, m – количество элементов в популяции, k_j – коэффициент штрафа, v_j – нарушение j-го ограничения, L – множество ограничений.

Функция $\bar{f}(\boldsymbol{x})$ определяется с учетом среднего значения целевой функции $\langle f(x) \rangle$ для текущей популяции:

$$\bar{f}(\boldsymbol{x}) = \begin{cases} f(\boldsymbol{x}) & \mid f(\boldsymbol{x}) < \langle f(\boldsymbol{x}) \rangle; \\ \langle f(\boldsymbol{x}) \rangle & \mid \langle f(\boldsymbol{x}) \rangle \le f(\boldsymbol{x}). \end{cases}$$

При этом коэффициент штрафа k_j, соответствующий j-му ограничению представлен в виде

$$k_j = \left| \langle f(x) \rangle \right| \frac{\langle v_j(x) \rangle}{\sum_{l=1}^{m} [\langle v_l(x) \rangle]^2},$$

где $\langle v_l(x) \rangle$ – нарушение l-го ограничения, усредненного для текущей популяции.

Таким образом, алгоритм ДЭ будет стремиться оптимизировать указанную целевую функцию $F(X)$, в результате этого он оптимизирует начальную целевую функцию, учитывая, например, ограничения (2), (3) в задаче о максимальном потоке.

[1] Ковалевич, А. А. Исследование стохастических алгоритмов оптимизации для применения в имитационном моделировании систем / А. А. Ковалевич, А. И. Якимов, Д. М. Албкеират // Информационные технологии. – 2011. – №8. – С. 55–60.

[2] Йенсен, П. Потоковое программирование: пер. с англ. / П. Йенсен, Д. Барнес. − М.: Радио и связь, 1984. – 392 с.: ил.

[3] Костевич, Л.С. Теория игр. Исследование операций: учеб. Пособие / Л.С. Костевич, А.А. Лапко. − Минск: Выш. школа, 1981. − 231 с.: ил.

[4] Barbosa, H. J. C. An Adaptive Penalty Scheme for Genetic Algorithms in Structural Optimization : H. J. C. Barbosa, A. C. C. Lemonge // Intnl. J. For Numerical Mehods in Engineering. − 2004. − №59. − P. 703−736.

[5] Silva, E. K. An Adaptive Constraint Handling Technique for Differential Evolution in Engineering Optimization / E. K. da Silva, H. J. C. Barbosa, A. C. C. Lemonge // EngOpt 2008 − International Conference on Engineering Optimization, Rio de Janeiro, Brazil, 01−05 June 2008. − Brazil, Rio de Janeiro, 2008. − P. 1−10.