Случайный поиск в задаче о назначениях

Выпускная квалификационная работа

Ставинова Елизавета Алексеевна

Санкт-Петербургский государственный университет
Прикладная математика и информатика
Вычислительная стохастика и статистические модели

Научный руководитель: д.ф.-м.н., профессор Сушков Ю.А.
Рецензент: к.ф.-м.н. Кушербаева В.Т.

СтатМод

Санкт-Петербург
2018г.

Введение

Задача оптимизации: $\Phi(x[1:n]) \to \min_x$, где Φ — ограниченная снизу функция, заданная на $[0;1]^n$.

Существует множество методов для решения данной задачи.

Одним из этих методов является метод случайного поиска. Он обладает рядом преимуществ:

- случайный поиск не предъявляет особых требований к оптимизируемой функции;
- алгоритм случайного поиска легко реализуем на ЭВМ.

В работе будет исследована эффективность алгоритма случайного поиска, а также данный алгоритм будет применен к решению задачи синтеза системы.

Алгоритм случайного поиска

Требуется: $\Phi(x[1:n]) \to \min_x$, где Φ — ограниченная снизу функция, заданная на $[0;1]^n$.

Обозначения:

- n_{step} — число шагов алгоритма;
- ϵ — точность, с которой ведется поиск;
- $I_i[k]$ — перспективная область для $x_i[k]$ (k — номер координаты, i — номер шага);
- $2q_i$ — ширина $I_i[k]$;
- a — параметр программы;
- $x_i^0[k]$ — центр $I_i[k]$;
- $p_i = P(x_i[k] \in I_i[k])$;
- $x_i^*[1:n]$ — текущий минимум за i шагов.

Распределение вероятностей, с которыми выбираются $x_i[k]$, принято равномерным как внутри, так и вне $I_i[k]$.

Алгоритм случайного поиска

Алгоритм:

1. $x_1^0[1:n] = 0.5,\ q_1 = 0.5, p_1 = 1;$
2. до тех пор, пока $i \leq n_{step}$:
   1. $q_i = q_{i-1}/a;$
   2. если $2q_i > \epsilon$, то $p_i = b|0.25 - q_i|^s - b \cdot 0.25^s + 1$, иначе $p_i = 1;$
   3. $x_i^0[1:n] = \max\{q_i; \min\{x_{i-1}^*[1:n]; 1 - q_i\}\};$
   4. моделируем α из распределения $\mathrm{U}[0;1]$: если $\alpha \leq p_i$, то моделируем $x_i[1:n]$ из распределения $\mathrm{U}(I_i[1:n])$, иначе моделируем $x_i[1:n]$ из распределения $\mathrm{U}([0;1]\backslash I_i[1:n]);$
   5. $x_i^*[1:n] = \begin{cases} x_i[1:n], & \text{если } \Phi(x_i[1:n]) < \Phi(x_{i-1}[1:n]), \\ x_{i-1}[1:n], & \text{если } \Phi(x_i[1:n]) \geq \Phi(x_{i-1}[1:n]). \end{cases}$

Логистическая кривая

Логистический закон изменения радиуса q_i перспективной области (Абакаров, Сушков, 2005):

$$q_i = 1 - \frac{1}{1 + (\frac{1}{V_0} - 1)e^{-\mu i / n_{step}}},$$

где μ — параметр крутизны логистической кривой (скорость изменения перспективной области), V_0 — первоначальный объем перспективной области.

Цель:

- нахождение оптимального параметра крутизны логистической кривой для функций из известного набора тестовых функций двумя способами;
- сравнение эффективности алгоритма при значениях параметров, полученных двумя способами.

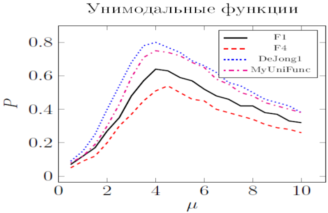
Классы тестовых функций

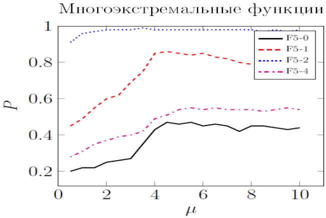
Исследования проводились на некоторых тестовых функциях из трех следующих классов:

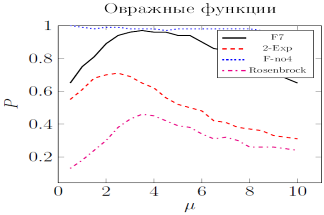
- Унимодальные функции (на множестве $X \subseteq \mathbb{R}^n$) — ровно одна точка локального минимума на этом множестве.
- Многоэкстремальные функции (на множестве $X \subseteq \mathbb{R}^n$) — более одного локального экстремума на этом множестве.
- Овражные функции (на множестве $X \subseteq \mathbb{R}^n$) — в исследуемой области собственные значения $\lambda_i(x)$ матрицы Гессе, упорядоченные в любой точке $x \in X$, удовлетворяют неравенствам: $0 < |\min\limits_i\{\lambda_i(x)\}| << \lambda_1(x), x \in X$.

Исследование эффективности алгоритма

Обозначим за P вероятность попадания в 0.02-окрестность минимума. Оптимальное значение параметра крутизны логистической кривой μ будем искать перебором по решетке.







Полученные результаты: для унимодальных тестовых функций $\mu_{\text{опт}} = 4$, для многоэкстремальных $\mu_{\text{опт}} = 5$, для овражных $\mu_{\text{опт}} = 3$.

Нахождение параметра $\mu_{\text{опт}}$ случайным поиском

Необходимо найти максимум функции зависимости P от μ, где P — вероятность попадания в заданную окрестность минимума, μ — параметр крутизны логистической кривой.

В работе был разработан следующий алгоритм:

Запустить случайный поиск с аргументом μ, промоделированным из $\mathrm{U}[0; 10]$, и функцией $\overline{P}$, полученной следующим образом:

1. выполнить (100 раз) случайный поиск для тестовой функции с μ, которое использовалось выше;
2. усреднить результаты пункта 1 (это и будет значение $\overline{P}$ — оценки функции P).

Были получены результаты:

Унимодальные		Многоэкстремальные		Овражные	
Функция	$\mu_{\text{опт}}$	Функция	$\mu_{\text{опт}}$	Функция	$\mu_{\text{опт}}$
F1	4.1685	F5-0	5.508	F7	2.8969
F4	4.3461	F5-1	4.8596	2-Exp	2.1979
DeJong1	3.9948	F5-2	6.0628	F-no4	0.3392
MyUniFunc	3.7675	F5-4	8.1858	Rosenbrock	3.7699

Сравнение результатов

- Обозначим за Δ разность между вероятностями попадания в заданную окрестность минимума для параметров, полученных перебором и случайным поиском соответственно.
- Для оценки бралось $\overline{\Delta}$ — усредненное значение Δ (количество запусков — 100), а также строился 95% доверительный интервал:

Унимодальные		Многоэкстремальные		Овражные	
Функция	$\overline{\Delta}$	Функция	$\overline{\Delta}$	Функция	$\overline{\Delta}$
F1	$0,01 \pm 0,013$	F5-0	$-0,007 \pm 0,014$	F7	$-0,006 \pm 0,004$
F4	$0,01 \pm 0,016$	F5-1	$0,005 \pm 0,01$	2-Exp	$0,03 \pm 0,012$
DeJong1	$0,005 \pm 0,012$	F5-2	$-0,002 \pm 0,004$	F-no4	$0,03 \pm 0,003$
MyUniFunc	$-0,003 \pm 0,01$	F5-4	$0,02 \pm 0,012$	Rosenbrock	$0,03 \pm 0,01$

Вывод:

- Оптимальные значения параметра крутизны логистической кривой, найденные случайным поиском, не дали значительных улучшений сходимости алгоритма.

Постановка задачи синтеза системы

Задано:

- $f_1(x[1:d]), \dots, f_n(x[1:d])$ — последовательность функций, где $f_i : D \in \mathbb{R}^d \to \mathbb{R}$ и $i \in I = \{1, \dots, n\}$
- $g[1], \dots, g[m]$ — последовательность чисел, $g[j] \in \mathbb{R}$ и $j \in J = \{1, \dots, m\}$
- $n \geq m$
- $\phi : J \to I$ — инъективное отображение
- $C(x[1:d], \phi) = \sum_{j \in J} (f_{\phi(j)}(x[1:d]) - g[j])^2$ — функция цели, отвечающая за приближение $f_{\phi(j)}(x[1:d])$ к $g[j]$

Задача синтеза системы:

$$C(x[1:d], \phi) \to \min_{x, \phi}$$

Цель: Изучение оптимизации с помощью случайного поиска функций данного вида.

Детерминированный поиск оптимального отображения

Рассмотрим следующий подход из работы Ю. А. Сушкова (2000).

- Целевая функция записывается таким образом:

$$C(x[1:d]) = \min_{\phi}(C(x[1:d], \phi)).$$

- Ищется минимум данной функции случайным поиском по аргументу $x[1:d]$.
- На каждом шаге случайного поиска находится отображение ϕ, которое минимизирует целевую функцию.
- Минимизирующее отображение достигается на множестве монотонных отображений $\phi : J \to I$.

Недостаток: время работы.

Случайный поиск оптимального отображения

Обратимся к другому подходу решения задачи:

- Введем отображение $\psi(y)$, которое переводит аргумент из единичного отрезка в одно из всевозможных отображений ϕ.
- Таким образом, функция цели принимает вид: $C(x[1:d], \psi(y)) = C(x[1:d], \phi)$.
- Применяем к функции цели случайный поиск по аргументам $x[1:d]$ и y.

В работе предложены и изучены 2 варианта задания отображения $\psi(y)$.

Способы задания отображения $\psi(y)$

Первый способ:

1. $i = \lceil y \cdot n! \rceil$ — это номер перестановки среди всех перестановок порядка n, перечисленных в лексикографическом порядке;
2. по i восстанавливается перестановка $(a_1, \ldots, a_n)$;
3. для задания отображения ϕ берутся первые m элементов $(a_1, \ldots, a_m)$.

Второй способ:

1. находятся всевозможные монотонные отображения, действующие из J в I;
2. для задания отображения ϕ берется отображение с номером $N = \lceil y \cdot C_n^m \rceil$.

Сравнение способов

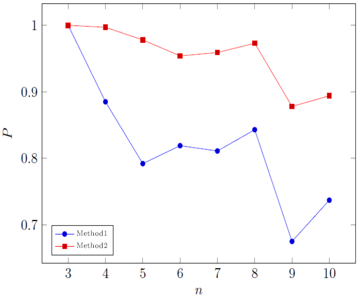


Рис. 1: Зависимость вероятности нахождения минимума целевой функции от размерности n вектор-функции; $n_{step} = 3000$, $m = 3$.

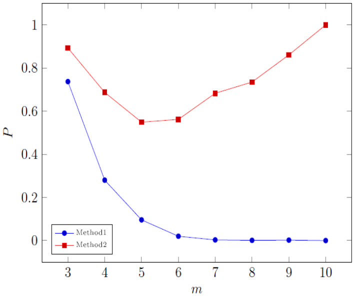


Рис. 2: Зависимость вероятности нахождения минимума целевой функции от размерности m вектора значений; $n_{step} = 3000$, $n = 10$.

Заключение

Результаты:

- Разработаны программы (в среде MATLAB):
  - для вычисления минимума заданной функции случайным поиском;
  - для нахождения оптимального параметра крутизны логистической кривой случайным поиском;
  - для решения задачи синтеза системы путем детерминированного поиска оптимального отображения;
  - для решения задачи синтеза системы с помощью случайного поиска.
- Решена задача нахождения оптимального значения параметра крутизны логистической кривой.
- Предложены два усовершенствования алгоритма решения задачи синтеза системы с помощью случайного поиска.
- Выявлен один более эффективный способ из рассмотренных двух.