

С. А. Крутман, В. Г. Поспехов

**МЕТОДЫ ГЛОБАЛЬНОЙ ОПТИМИЗАЦИИ
ОПТИЧЕСКИХ СИСТЕМ**

Приведен обзор некоторых современных методов глобальной оптимизации оптических систем, в частности реализованных в крупнейших оптических САПР. Рассмотрены принципы и алгоритмы, лежащие в основе этих методов.

E-mail: minus5@yandex.ru

Ключевые слова: глобальная оптимизация, расчет оптических систем, оптические САПР, генетические алгоритмы, алгоритм имитации отжига.

Одной из основ современного расчета оптических систем (ОС) является их автоматизированная оптимизация. В отличие от широко используемых методов локальной оптимизации, существенным недостатком которых является “застревание” в первом же найденном минимуме, глобальная оптимизация в пределе позволяет найти решение с наименьшим значением целевой функции на всем пространстве допустимых значений параметров системы. При этом в то время как применяемые в оптике методы локальной оптимизации подробно описаны в литературе [1–4], методы глобальной оптимизации рассмотрены лишь в отдельных зарубежных статьях [5–19], что препятствует пониманию и оптимальному использованию глобальных оптимизаторов, предоставляемых современными оптическими САПР (CodeV, OSLO, Zemax). В связи с этим цель настоящей работы — описание принципов, лежащих в основе наиболее заметных методов глобальной оптимизации, используемых при расчете ОС.

Среди методик, используемых для реализации глобальной оптимизации применительно к расчету ОС, следует отметить глобальный поиск [5], оптимизацию с использованием эскейп-функции [6, 7], алгоритмы имитации отжига [8–10], генетические алгоритмы [11–15], алгоритмы на основе методов седловых точек [16, 17], глобальный синтез [18, 19]. Далее остановимся на первых четырех методиках как наиболее знаковых.

Наиболее простой способ глобальной оптимизации, очевидно, — метод прямого перебора. Примером такого подхода является метод глобального поиска, предложенный Д. О’Ши в работе [5].

Алгоритм представляет собой поиск в два этапа (рис. 1 [5]). На первом этапе запускается алгоритм разбиения, с помощью которого на всем многомерном пространстве поиска вводится грубая сетка путем разбиения области изменения каждого из параметров на некоторое

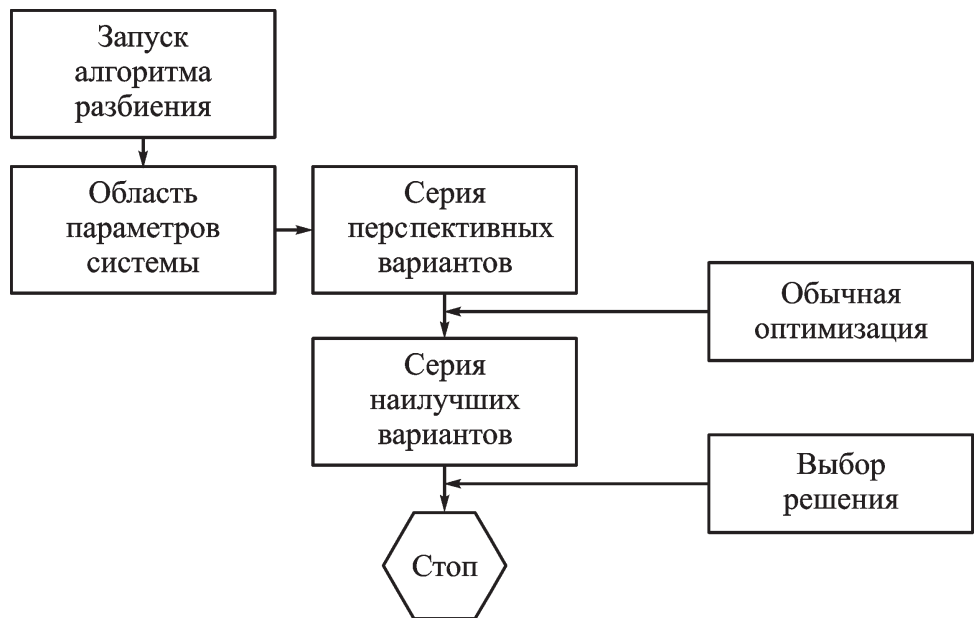


Рис. 1. Блок-схема алгоритма глобальной оптимизации по методу глобального поиска

число интервалов. Оптические системы, параметры которых соответствуют узлам полученной сетки, оцениваются с использованием простейшей оценочной функции на основе аберраций третьего порядка вида

$$MF = \left[\sum_{i=1}^5 (S_i)^2 + \sum_{j=1}^2 (C_j)^2 \right]^{1/2},$$

где S_i — монохроматические суммы Зейделя; C_j — хроматические суммы. На основе предварительной оценки отбираются от 10 до 100 перспективных вариантов ОС и проводится их предварительная оптимизация.

На втором этапе из отобранных и оптимизированных систем выбирается серия наилучших вариантов ОС и проводится их окончательная оптимизация с использованием точной целевой функции на основе реальных пятен рассеяния

$$MF = \left[\sum_{i=1}^3 \Delta_i^2 \right]^{1/2},$$

где Δ_i — среднеквадратическое значение радиуса полихроматического пятна рассеяния для трех значений угла поля.

В целях улучшения методики и распространения ее на расчет сложных ОС авторы [5] внесли в базовую процедуру ряд дополнений, ускоряющих работу алгоритма на первом этапе, в частности процедуру скрининга, оптимизацию сетки и использование параллельных вычислений.

Роль процедуры скрининга состоит в экономии вычислительного времени благодаря заблаговременному отсечению сочетаний параме-

тров ОС, не имеющих физического смысла, и отсутствия необходимости проводить связанные с этими ОС вычисления. Для этого в алгоритм вводится ряд эвристических правил отбраковки параметров. К таким правилам, например, относятся следующие:

— относительная оптическая сила каждого элемента должна быть как можно меньше, а относительное отверстие элементов не должно превышать определенных значений;

— максимальное по модулю значение кривизны поверхности не должно превышать предельного значения, геометрически ограниченного ее апертурой.

Под оптимизацией сетки понимается введение сетки на первом этапе с учетом влияния параметров ОС на целевую функцию. Так, область значений параметров, оказывающих наибольшее влияние на целевую функцию, разбивается чаще, т.е. на мелкие интервалы, в то время как область значений параметров с меньшим влиянием на целевую функцию разбивается на большие интервалы или такие параметры вообще фиксируются.

Такая методика, по утверждениям ее авторов, применима к расчету ОС с размерностью пространства поиска до 15.

Метод прямого перебора, подобный описанному, хотя и является самым верным путем определения глобального минимума, на практике оказывается малоприменимым, так как требует огромного времени расчета в силу так называемого проклятия размерности. Предположим, что требуется найти глобальный минимум для простейшей четырехлинзовой системы со всеми сферическими поверхностями и фиксированными свойствами оптических сред. С учетом требования сохранения заданного значения фокусного расстояния такая система имеет 15 варьируемых параметров: радиусы поверхностей и толщины. Даже в случае очень грубого и малоинформативного разбиения диапазона изменения каждого параметра на 10 интервалов, получается 10^{15} систем и, даже если для (очень грубой) оценки качества через каждую систему требуется просчитать 10 лучей, решение задачи глобальной оптимизации методом перебора сводится к расчету $8 \cdot 10^{16}$ лучей через поверхность, что с учетом среднего быстродействия современных доступных компьютеров 10^8 лучей через поверхность в секунду [20] требует $\sim 8 \cdot 10^8$ с, т.е. приблизительно 25 лет.

При этом разбиение диапазонов изменения конструктивных параметров хотя бы на 30 точек увеличивает это время до ~ 360 млн лет! Ясно, что при таких временных затратах даже на грубый расчет элементарной системы такой метод прямого перебора для решения задачи глобальной оптимизации становится абсолютно неприменим.

Простейшей в реализации альтернативной методу прямого перебора является оптимизация с использованием эскейп-функции (или функции ухода), предложенная М. Ишики [6, 7]. Методика не требует внесения принципиальных изменений в имеющиеся локальные оптимизаторы и основана на добавлении к исходной функции ошибки на определенных этапах эскейп-функции специального вида, заставляющей оптимизатор выходить из локальных минимумов.

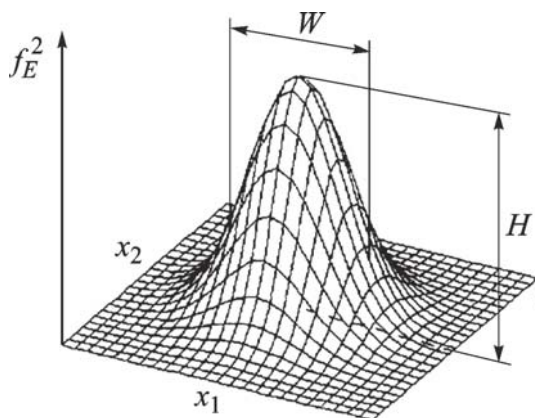


Рис. 2. Приращение целевой функции в районе локального минимума при добавлении эскейп-функции

Эскейп-функция, предложенная М. Ишики, определяется как

$$f_E = \sqrt{H} \exp \left[-\frac{1}{2W^2} \sum_j \{ \mu_j (x_j - x_{jL}) \}^2 \right],$$

где x_j — j -й параметр оптической системы; x_{jL} — значение j -го параметра оптической системы в точке локального минимума; μ_j — коэффициент масштаба для j -го параметра. При добавлении функции, соответствующей данному описанию, к исходной функции ошибки новая целевая функция возрастает в районе локального минимума, при этом приращение равно f_E^2 и имеет вид, приведенный на рис. 2 [6]. Дальнейшая оптимизация уводит решение из локального минимума, а параметры эскейп-функции H и W автоматически настраиваются и регулируются внешним программным модулем для улучшения сходимости и выхода из минимума в соответствии с текущим рельефом целевой функции.

Оптимизация при использовании эскейп-функции проводится по алгоритму, приведенному на рис. 3 [6]. При этом в качестве критерия выхода решения из локального минимума используется расстояние между рассматриваемым локальным минимумом и новым решением, определяемое как

$$D_p = \sqrt{\sum_j \{ \mu_j^2 (x'_j - x_j)^2 \}},$$

где x_j, x'_j — координаты j -го параметра системы для рассматриваемого локального минимума и нового решения.

В случае, если это расстояние больше заданного расчетчиком, считается, что система успешно вышла из локального минимума, в противном случае выход признается неудачным, а полученная система — слишком близкой по своим параметрам к предыдущему полученному

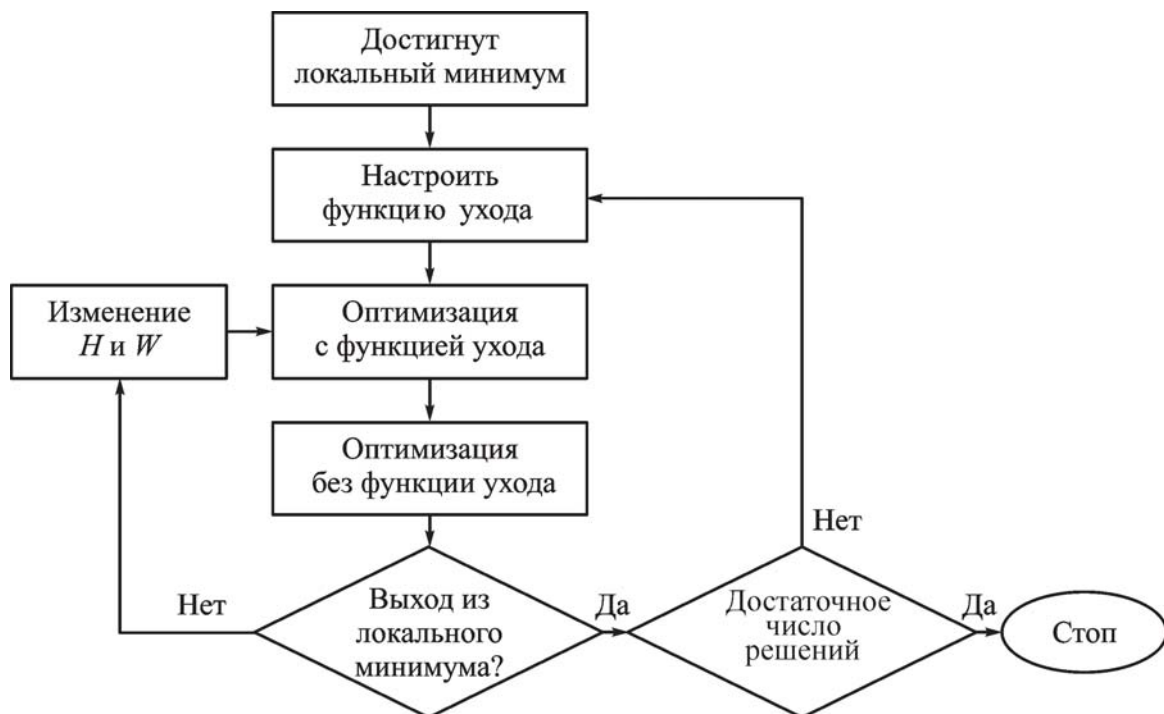


Рис. 3. Блок-схема алгоритма М. Ишики глобальной оптимизации при использовании эскейп-функции

варианту, проводится изменение параметров эскейп-функции W и H и попытка повторяется.

В качестве алгоритма для изменения параметров эскейп-функции в случае неудачного выхода из локального минимума в работе [6] предлагается использовать следующее эмпирическое правило:

$$\begin{aligned}
 H_1 &= 2H_0, & W_1 &= (1 + 0,03 \cdot 1) W_0; \\
 H_2 &= 2^2 H_0, & W_2 &= (1 + 0,03 \cdot 2) W_0; \\
 &\dots\dots\dots & & \\
 H_k &= 2^k H_0, & W_k &= (1 + 0,03 \cdot k) W_0.
 \end{aligned}$$

Несомненным достоинством методики Ишики является простота реализации при достаточно высокой эффективности. Отметим, что эта методика составляет основу одного из алгоритмов глобальной оптимизации Global Explorer оптической САПР OSLO [21].

Одним из альтернативных направлений в глобальной оптимизации ОС являются методы, копирующие различные природные процессы, например: алгоритмы имитации отжига [8–10], копирующие процесс, происходящий при медленном охлаждении вещества, и генетические алгоритмы [11–15], имитирующие появление новых особей (решений) и эволюционный отбор лучших из них. Далее подробно остановимся на этих двух видах алгоритмов глобальной оптимизации, получивших широкое применение в оптических САПР.

В 1983 г. С. Киркпатрик [22] указал на схожесть процесса оптимизации и процесса получения низкотемпературного состояния вещества при отжиге, предложив реализовывать первое как имитацию второ-

го, уподобив различные перестройки оптимизируемой системы перестройкам системы частиц охлаждаемого вещества, а оптимизируемую функцию — энергии системы частиц этого вещества. При этом основой подхода была принята имитация ключевого свойства процесса отжига, гарантирующего в пределе получение системы с минимально возможной энергией и заключающегося в том, что в состоянии теплового равновесия (из которого система при отжиге практически не выходит) допускается с некоторой вероятностью существование конфигураций с энергией E , превышающей энергию некоторого текущего рассматриваемого состояния (при этом вероятность существования конфигурации с энергией E равна $\sim \exp(-E/(k_B T))$, где k_B — постоянная Больцмана, T — температура). Другими словами, при оптимизации должны с некоторой вероятностью допускаться шаги, ухудшающие целевую функцию, причем вероятность принятия таких шагов постепенно уменьшается с уменьшением “температуры”. Для имитации этого ключевого свойства С. Киркпатрик использовал так называемый алгоритм Метрополиса [23], предложенный М. Розенблютом, Н. Метрополисом и другими, для моделирования равновесных состояний систем атомов при заданных температурах.

Алгоритм имитации отжига для оптимизации функции, таким образом, сводится к следующим шагам.

1. “Нагрев” системы, т.е. присваивание температуре T начального высокого значения.

2. Выбор начальной базовой точки $X_b = (x_{b1}; x_{b2} \dots x_{bN})$ и вычисление начального значения целевой функции $f_b = f(X_b)$.

3. Выбор случайного шага $S = (s_1; s_2 \dots s_N)$ из распределения, называемого генератором, и получение пробной точки $X_t = X_b + S$, вычисление $f_t = f(X_t)$.

4. Сравнение текущего значения целевой функции с начальным значением:

— если $f_t < f_b$, пробная точка принимается, f_b и X_b присваиваются значения f_t и X_t соответственно;

— если $f_t > f_b$, пробная точка принимается с вероятностью $\exp\left[-\frac{1}{T}(f_t - f_b)\right]$.

5. Обновление параметров: присваивание новых значений температуре T и параметрам генератора.

6. Проверка критерия остановки. Если критерий не удовлетворен, возврат к шагу 3.

Легко видеть, что алгоритм имитации отжига в основе своей достаточно прост. Однако за пределами этой основы существуют два важных и далеко не столь простых аспекта, определяющие скорость

получения результата и саму возможность получения результата за реальное время. Такими аспектами являются оптимальное расписание отжига, т.е. режим понижения температуры, и свойства генератора, определяющие оптимальность пробных шагов.

На сегодняшний день разработано множество модификаций алгоритма имитации отжига. Далее остановимся на одном из вариантов, разработанном Г. Форбсом и Э. Джонсом специально для нужд проектирования ОС, так называемом алгоритме адаптивной имитации отжига (АИО) [8–10].

Вслед за авторами будем использовать три основных понятия:

– плотность распределения $p_i(X)$ определяется таким образом, что произведение $p_i(X) dV_X$ равно вероятности нахождения базовой точки после итерации i в объеме dV_X в районе точки X , соответственно

$$\int_{\Omega} p_i(X) dV_X = 1;$$

– генератор $g(Y, X, \Psi)$ определяется так, что $g(Y, X, \Psi) dV_Y$ равно вероятности генерации пробной точки в объеме dV_Y в районе точки Y при условии, что базовая точка имеет координату X , при этом Ψ обозначает совокупность свойств генератора;

– функция принятия шага $a(Y, X, \beta)$, определяемая как вероятность принятия перехода из точки X в точку Y , при данном значении обратной температуры $\beta = \frac{1}{T}$ имеет вид

$$a(Y, X, \beta) = \begin{cases} 1, & f(Y) < f(X), \\ e^{-\beta[f(Y)-f(X)]}, & f(Y) > f(X) \end{cases} = e^{-\beta \cdot \max[0, (f(Y)-f(X))]}.$$

Используя приведенные понятия, один цикл алгоритма имитации отжига можно смоделировать следующим образом:

$$p_{i+1}(X) = p_i(X) + \int_{\Omega} p_i(Y) a(X, Y, \beta) g(X, Y, \Psi) dV_Y - \\ - \int_{\Omega} -p_i(X) a(Y, X, \beta) g(Y, X, \Psi) dV_Y,$$

где первый интеграл обозначает увеличение плотности распределения за счет вероятности прихода базовой точки в район точки X , а второй интеграл обозначает уменьшение плотности распределения вследствие вероятности ухода базовой точки из района точки X .

В силу природы алгоритма для любых фиксированных значений β и Ψ с ростом числа шагов i распределение $p_i(X)$ стремится к некоторому фиксированному распределению, называемому равновесным распределением $\pi(X, \beta, \Psi)$, и определяемому из условия

$p_{i+1}(X) = p_i(X) = \pi(X, \beta, \Psi)$, что при условии симметричности генератора $g(Y, X, \Psi) = g(X, Y, \Psi)$ дает

$$\pi(X, \beta) = \alpha(\beta) e^{-\beta f(X)},$$

где $\alpha(\beta) = \left[\int_{\Omega} e^{-\beta f(Y)} dV_Y \right]^{-1}$.

Видно, что при $\beta = 0$ ($T \rightarrow \infty$) равновесное распределение стремится к равномерному, по мере же увеличения β (уменьшения T) равновесное распределение становится плотнее в районах, где целевая функция меньше, и, наконец, в пределе, при высоких β распределение полностью сосредотачивается в районе глобального минимума. Таким образом, бесконечно медленное в пределе увеличение β до ∞ при многократном повторении циклов, так что распределение не отклоняется от равновесного распределения, гарантирует нахождение глобального минимума. На практике бесконечно долгое время работы алгоритма недопустимо, и в случае алгоритма АИО расписание отжига и свойства генератора формируются, исходя из следующего принципа: повышать β как можно быстрее, при этом пытаясь сохранить плотность распределения в пределах определенного (заданного) расстояния от плотности равновесного распределения.

Тогда если в качестве меры расстояния между двумя плотностями распределения p' и p'' используется величина

$$D[p', p''] = \frac{1}{2} \int_{\Omega} |p'(Y) - p''(Y)| dV_Y,$$

а обратная температура β на m -м цикле равна β_m , то положенный в основу алгоритма АИО принцип может быть записан как

$$D[p_m(X), \pi(X, \beta_m)] < \varepsilon \quad \forall m,$$

где величина ε задается расчетчиком и является своего рода мерой тщательности работы алгоритма. В соответствии с этим критерием для расписания отжига в алгоритме АИО выведено следующее выражение, позволяющее рассчитать $\Delta\beta$ на каждом цикле алгоритма:

$$\Delta\beta = \frac{\varepsilon(1 - R')}{S}, \tag{1}$$

где S — так называемая чувствительность, являющаяся мерой смещения равновесного распределения при изменении обратной температуры и определяемая как

$$S = \lim_{\Delta\beta \rightarrow 0} \left\{ \frac{D[\pi(X, \beta), \pi(X, \beta + \Delta\beta)]}{\Delta\beta} \right\}, \tag{2}$$

а R' — фактор сближения, являющийся мерой скорости восстановления равновесного состояния системы, следующего вида:

$$R' = \lim_{\Delta\beta \rightarrow 0} \left\{ \frac{D[p_{i+1}(X), \pi(X, \beta + \Delta\beta)]}{D[p_i(X), \pi(X, \beta + \Delta\beta)]} \right\}. \quad (3)$$

Таким образом, выражения (1)–(3) определяют адаптивный режим отжига на каждом цикле.

При этом второй ключевой аспект любого алгоритма имитации отжига — свойства Ψ генератора пробных шагов $g(X, Y, \Psi)$ — в алгоритме АИО определяют, исходя из критерия максимальности скорости уравнивания системы $R = 1 - R'$. В качестве базовой формы здесь предлагается гауссов генератор с матрицей ковариации C :

$$g(Y, X, C) = \frac{1}{\sqrt{(2\pi)^N \det C}} e^{-\frac{1}{2}(Y-X) \cdot C^{-1}(Y-X)}.$$

Тогда свойства Ψ генератора определяются элементами матрицы ковариации C .

В случае слишком длинных пробных шагов вероятность принятия шага мала, и скорость минимизации падает. При слишком малых пробных шагах вероятность принятия шага велика, однако в силу малости размера шага скорость минимизации опять замедляется. Таким образом, можно говорить о существовании некоторого оптимального размера шага. В алгоритме АИО используется анизотропный гауссов генератор, адаптируемый на каждом шаге к изменению рельефа целевой функции. Для этого, опираясь на центральную предельную теорему, авторы алгоритма предлагают определять пробный шаг на основе случайно взвешенной суммы принятых на предыдущих этапах шагов r_m :

$$s = \frac{\rho\nu\xi}{\sqrt{M}} \sum_{m=1}^M \omega_m \bar{r}_m,$$

где M — число использованных последних принятых шагов ($M \sim 10N$, где N — размерность пространства); ω_m — случайный вес из распределения с нулевым математическим ожиданием и единичной дисперсией; ρ, ν — коэффициенты оптимального масштаба (больше единицы), учитывающие тот факт, что в силу более высокой вероятности отклонения длинных пробных шагов принятые шаги, используемые для генерации очередного пробного шага, обычно короче генерируемых шагов; ξ — коэффициент, учитывающий отклонение реальной целевой функции от квадратичной модели, используемой для определения ρ и ν , при этом \bar{r}_m — принятые пробные шаги, разделенные на ξ в момент их принятия.

Здесь приведены лишь основные моменты алгоритма АИО, наиболее детальное описание представлено в работе [10]. Отметим, что указанный алгоритм заслуживает особого внимания в том числе и потому, что подобно предыдущему также нашел практическое применение во второй по мощности оптической САПР OSLO (алгоритм ASA) [21].

Еще одним способом глобальной оптимизации, применяемым в задачах расчета ОС, являются генетические алгоритмы [11–14]. В частности, такие алгоритмы являются основой двух глобальных оптимизаторов — Global Search и Hammer, используемых в оптической САПР Zemax [24].

Генетический алгоритм [25] — это эвристический алгоритм поиска, используемый для решения задач оптимизации с использованием механизмов, имитирующих биологическую эволюцию. При этом в случае генетического алгоритма под эволюцией подразумевается эволюция некоторой популяции особей (хромосом)–решений, приспособленность каждой из которых определяется значением целевой функции, соответствующим данному решению. В простейшем случае канонического генетического алгоритма, блок-схема которого представлена на рис. 4 [11], имитация такой эволюции сводится к имитации появления новых особей-потомков (новых решений) на основе скрещивания особей-родителей (старых решений), имитации отбора наиболее приспособленных особей (решений с наилучшим значением целевой функции) и имитации случайных мутаций (редких случайных изменений решений).

На начальном этапе $n = 0$ канонического генетического алгоритма случайно генерируется начальная популяция хромосом, каждая из которых представляет собой последовательность генов, кодирующих альтернативное решение (например, хромосома может кодировать вариант ОС, при этом каждый ген может нести в себе значение соответствующего конструктивного параметра ОС). Затем стартует цикл, на каждой итерации которого к текущей популяции последовательно применяются: *оператор репродукции*, случайно отбирающий хромосомы для скрещивания с вероятностью, пропорциональной их функции приспособленности (определяемой значениями целевой функции соответствующих ОС); *оператор кроссинговера*, имитирующий образование хромосом-потомков, заимствующих отдельные участки генетического кода у родителей (образование новых ОС, наследующих различные группы конструктивных параметров у различных отобранных ранее старых ОС); *оператор случайной мутации*, с заданной (малой) вероятностью изменяющий хромосому в случайном месте случайным образом; и, наконец, *оператор рекомбинации*, определяющий хромосомы, которые войдут в следующую популяцию (отбирающий наиболее целесообразные для дальнейшей эволюции ОС в соответствии с

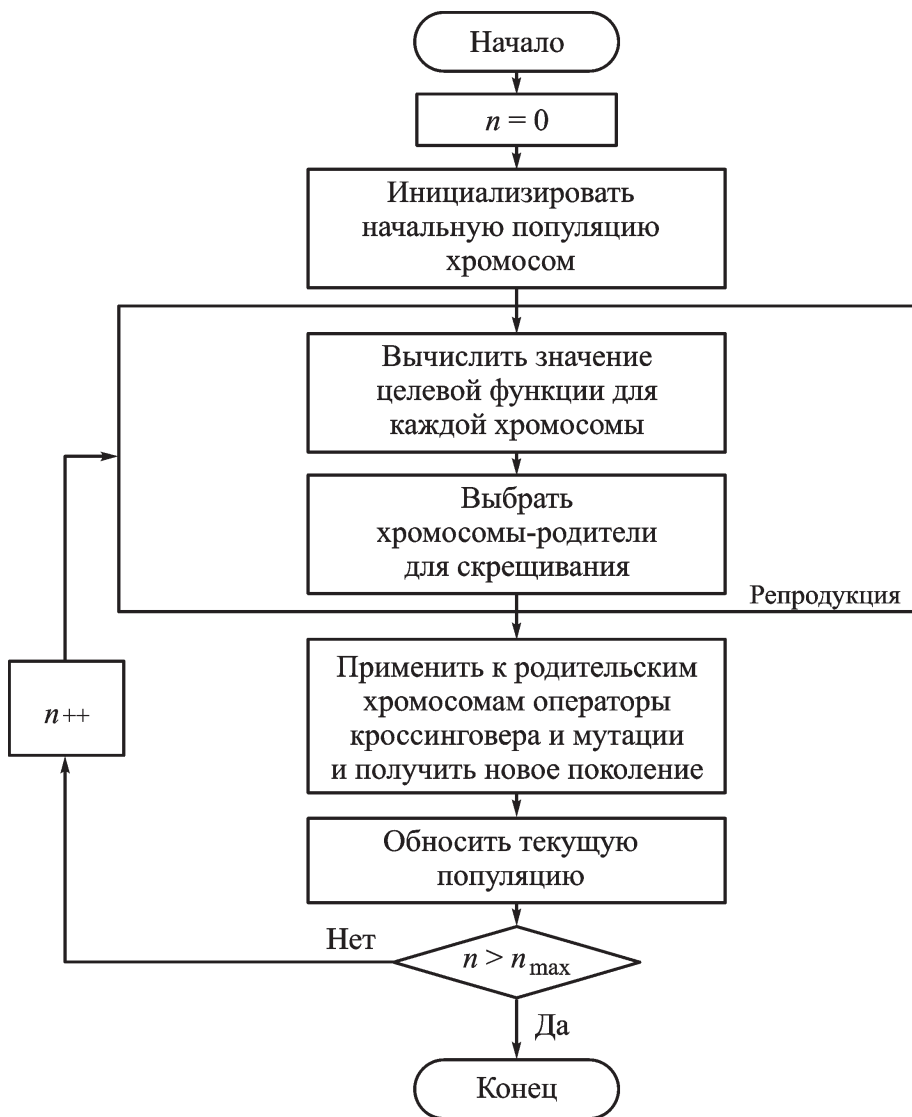


Рис. 4. Блок-схема канонического генетического алгоритма

их целевой функцией). Цикл продолжается до тех пор, пока не будет достигнуто максимальное число итераций n или получено удовлетворительное решение.

Применение генетических алгоритмов в каноническом и немного модифицированном виде к задаче расчета и оптимизации ОС представлено, например, в работах [11–13]. В работах [11, 12] К. Ямамото описывает пример использования генетического алгоритма для расчета семилинзового объектива с фокусным расстоянием 28 мм, относительным отверстием 1:2,8 и угловым полем 75° . При этом генетический алгоритм используется для поиска начальной, грубой конфигурации, а для окончательной оптимизации используется традиционный метод наименьших квадратов. Используемый К. Ямамото генетический алгоритм обладал следующими характеристиками: размер популяции $n = 3000$ хромосом; длина хромосомы $l = 41$ (14 радиусов, 13 толщин, 7 марок стекол (n_d, ν_d) , рис. 5 [11]); вероятность кроссинговера $r_c \in (0,75–0,98)$; вероятность мутации $r_m \in (0,02–0,25)$; максимальное число итераций — 1000.

Радиус 1	Радиус 2	...	Толщ 1	Толщ 2	...	n_d1	n_d2	...	v_d1	v_d2	...
----------	----------	-----	--------	--------	-----	--------	--------	-----	--------	--------	-----

Рис. 5. Структура хромосомы в примере К. Ямамото

В качестве целевой функции, позволяющей в процессе работы генетического алгоритма наряду с оптимизацией качества изображения обеспечить контроль основных геометрических параметров, К. Ямамото предложил использовать двухступенчатую функцию вида

$$MF = \begin{cases} MF_A; \\ MF_A + \Delta, & MF_A < \text{const}, \end{cases}$$

где MF_A — часть, отвечающая за обеспечение основных геометрических параметров ОС и представляющая собой сумму взвешенных, взятых по модулю отклонений текущих значений таких параметров, как фокусное расстояние, высоты лучей осевого и наклонного пучков на входной плоскости ОС и других от требуемых, а Δ — часть, характеризующая качество изображения и учитываемая лишь для ОС, значения геометрических параметров которых находятся вблизи требуемых.

На этапе репродукции вероятность P_i отбора для скрещивания для каждой хромосомы популяции вычислялась как [12]

$$P_i = \frac{MF_{\min}}{MF_i},$$

где MF_i — значение целевой функции для соответствующей хромосомы; MF_{\min} — минимальное значение целевой функции в популяции. При этом хромосома отбиралась для скрещивания (или мутации) в случае если случайное число, сгенерированное в интервале от 0 до 1, оказывалось меньше P_i .

Существует и множество других схем селекции [25], наиболее популярной из которых является селекция на основе рулетки, когда каждой хромосоме на колесе рулетки соответствует зона, размер которой пропорционален отношению функции приспособленности хромосомы к сумме значений функции приспособленности всех хромосом популяции. Таким образом, при вращении колеса рулетки более приспособленные хромосомы имеют больше шансов быть отобранными для дальнейшего скрещивания.

На этапе кроссинговера использовались операторы одно-, двух- и многоточечного кроссинговера. При одноточечном кроссинговере две хромосомы-потомки образовывались от двух хромосом-родителей путем разрезания последних в одинаковых случайных точках на две части и обмена аналогичными частями. При двухточечном кроссинговере случайно выбираются две точки разреза, и хромосомы-потомки

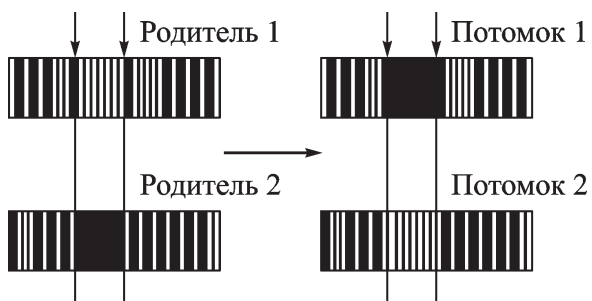


Рис. 6. Схема двухточечного кроссинговера

образуются при обмене хромосом-родителей участками, находящимися между выбранными точками (рис. 6 [11]). Многоточечный кроссинговер строится по аналогии с двухточечным для большего числа случайных точек разреза.

Оператор мутации в интерпретации К. Ямамото присваивает случайному биту хромосомы случайное значение. Существует множество других версий оператора мутации [25], например перестановка двух соседних битов, двух случайно выбранных битов или целых участков хромосом и др. Роль мутации в генетическом алгоритме, по аналогии с биологической эволюцией, состоит в поддержании и внесении генетического разнообразия в популяцию, что предотвращает преждевременную сходимость оптимизации к локальным минимумам. Для решения проблемы замедления сходимости в своем алгоритме К. Ямамото использует адаптивный контроль вероятности мутации, при котором вероятность мутации увеличивается при падении скорости сходимости. Также в случае преждевременной сходимости для выхода из локального минимума в алгоритме используется имитация катастрофы, когда каждая хромосома в популяции подвергается некоторому случайному изменению.

При решении К. Ямамото описанной выше задачи о синтезе объектива для начальной популяции из 3000 случайно сгенерированных хромосом среднее значение целевой функции равнялось 152,533, наименьшее значение целевой функции было равно 1,477. После 658 итераций представленного генетического алгоритма целевая функция значительно уменьшилась и стала равна 0,087 в среднем по популяции и 0,059 для наилучшего решения, представленного после окончательной оптимизации методом наименьших квадратов на рис. 7 [11].

Следует отметить, что для получения представленного решения с помощью генетического алгоритма не требовалось задавать никаких начальных точек, только саму общую структуру (семь одиночных линз). Это является несомненным достоинством применения генетических алгоритмов.

Однако Э. Бетенский пошел еще дальше и в своей работе [15] предложил использовать генетический алгоритм не просто для синтеза и оптимизации ОС известной структуры, но для синтеза самой этой структуры.

Фактически, можно сказать, что подход Э. Бетенского сводится к имитации работы оптика-расчетчика с помощью генетического

алгоритма. Для этого автор использует концепцию так называемых операторов структурных изменений, представляющих собой типичные модификации, вносимые оптиком в систему в процессе оптимизации и не влияющие существенно на гауссовы характеристики системы (например, изгиб линзы или элемента, разворот линзы или элемента и др.).

Схематично представляя работу дизайнера ОС как последовательное внесение структурных изменений-гипотез в систему и ее оптимизацию-проверку, повторяемые многократно, Э. Бетенский предлагает кодировать совокупности различных структурных изменений в виде списка операторов в хромосомы и, применяя генетический алгоритм, находить оптимальную совокупность структурных изменений исходной системы, т.е. фактически, новую оптимальную структуру.

Предоставляя разработчику уникальную возможность автоматизированного синтеза ОС на уровне структуры, методика Э. Бетенского при должной реализации несомненно включает в себе огромный потенциал.

Таким образом, приведен обзор некоторых ключевых методов глобальной оптимизации, используемых в расчете ОС, часть которых составляет основу алгоритмов глобальных оптимизаторов крупнейших оптических САПР — OSLO и Zemax. Метод прямого перебора в чистом виде в целях глобальной оптимизации на практике не применяется из-за крайне медленной его работы уже в случае простейших ОС. Применение методов оптимизации при использовании эскейп-функции, генетических алгоритмов и алгоритма имитации отжига для простых ОС дает достаточно хороший результат, однако, в случае сложных ОС, например современных панкратических систем с размерностью пространства переменных целевой функции более 150, эти методы малоэффективны. В таких случаях представляется целесообразным использование блочно-модульного подхода, позволяющего проектировать ОС как совокупность простых подсистем, которые, в свою очередь, могут быть получены с использованием описанных методов. Понимание принципов, лежащих в основе наиболее применяемых в вычислительной оптике методов глобальной оптимизации, дает возможность более эффективно работать с современными оптическими САПР, а также позволяет самостоятельно реализовывать указанные методики применительно к собственным программам и задачам.

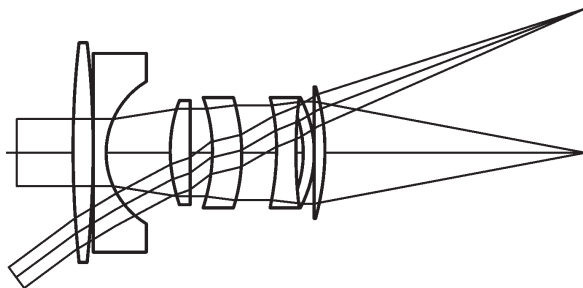


Рис. 7. Наилучший вариант объектива, полученный с помощью генетического алгоритма и последующей оптимизации методом наименьших квадратов

СПИСОК ЛИТЕРАТУРЫ

1. Родинон С. А. Автоматизация проектирования оптических систем. – Л.: Машиностроение, 1982. – 270 с.
2. Слюсарев Г. Г. Методы расчета оптических систем. – Л.: Машиностроение, 1969. – 672 с.
3. Kinglake R., Barry Johnson R. Lens design fundamentals // Academic Press. – 2010. – 549 p.
4. Shannon R. R. The art and science of optical design. – Cambridge University Press. – 1997. – 611 p.
5. Doron Sturlesi, Donald C. O'Shea. Global view of optical design space // Optical Engineering. – 1991. – Vol. 30, 207.
6. Masaki Isshiki. Global optimization with escape function // Proc. SPIE 3482. – 1998. – Vol. 104.
7. Akira Yabe. Global optimization of zoom lenses // Proc. SPIE 3482. – 1998. – Vol. 122.
8. Forbes G. and Jones A. Global optimization in lens design // Optics & Photonics News. – 1992. – Vol. 3 (3). – P. 22–29.
9. Forbes G. W. and Jones A. E. W. Towards global optimization with adaptive simulated annealing // Proc. SPIE 1354. – 1991. – P. 144–151.
10. Jones A. E. W. and Forbes G. W. An adaptive simulated annealing algorithm for global optimization over continuous variables // J. of Global Optimization. – 1995. – Vol. 6. – P. 1–37.
11. Xiaogang Chen, Kimiaki Yamamoto. Genetic algorithm and its application in lens design // Proc. SPIE 2863. – 1996. – P. 216.
12. Xiaogang Chen, Kimiaki Yamamoto. An experiment in genetic optimization in lens design // J. of Modern Optics. 1997. – Vol. 44. No. 9. – P. 1693–1702.
13. Kenneth E. Moore. Algorithm for global optimization of optical systems based on genetic competition // Proc. SPIE 3780. – 1999. – Vol. 40.
14. Isao Ono, Shigenobu Kobayashi, and Koji Yoshida. Global and multi-objective optimization for lens design by real-coded genetic algorithms // Proc. SPIE 3482. – 1998. – P. 110.
15. Betensky Ellis I. Postmodern lens design // Optical Engineering. – 1993. – Vol. 32. – 1750 p.
16. Florian Bociort, Eco van Driel, Alexander Serebryakov. Networks of local minima in optical system optimization // Optics Letters. – 2004. – Vol. 29. No. 2.
17. Florian Bociort, Maarten van Turnhout. Finding new local minima in lens design landscapes by constructing saddle points // Optical Engineering. 2009. – Vol. 48 (6).
18. Thomas G. Kuper, Thomas I. Harris. Global optimization for lens design: An emerging technology // Proc. SPIE 1781. 1993. – Vol. 14.
19. Thomas G. Kuper, Thomas I. Harris. Practical strategy for global optimization of zoom lenses // Proc. SPIE 3482. – 1998. – Vol. 135.
20. The ZEMAX Newsletter, spring 2010.
21. O S L O Optics Reference / Lambda Research Corporation, June 2005. – 429 p.
22. Kirkpatrick S., Gelatt C. D., Vecchi J. r. M. P. Optimization by simulated annealing // Science. – 1983. – Vol. 220. No. 4598. – P. 671–680.
23. Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, Edward Teller. Equation of state calculations by fast computing machines // Journal of Chemical Physics. – 1953. – Vol. 21. No. 6. – P. 1087–1092.

24. И н ф о р м а ц и я с сайта <http://www.zemax.com> на 30.03.2011.

25. Г л а д к о в Л. А., К у р е й ч и к В. В., К у р е й ч и к В. М. Генетические алгоритмы. – М.: Физматлит, 2010. – 368 с.

Статья поступила в редакцию 18.04.2011

Семен Александрович Крутман родился в 1982 г., окончил в 2006 г. МГТУ им. Н.Э. Баумана. Аспирант кафедры “Оптико-электронные приборы научных исследований” МГТУ им. Н.Э. Баумана. Специализируется в области расчета оптических систем.

S.A. Krutman (b. 1982) graduated from the Bauman Moscow State Technical University in 2006. Post-graduate of “Optoelectronic Research Devices” department of the Bauman Moscow State Technical University. Specializes in the field of optical system design.

Вячеслав Георгиевич Поспехов родился в 1937 г., окончил в 1961 г. МВТУ им. Н.Э. Баумана. Канд. техн. наук, доцент кафедры “Оптико-электронные приборы научных исследований” МГТУ им. Н.Э. Баумана. Автор около 60 научных работ в области оптики, расчета оптических систем.

V.G. Posphehov (b. 1937) graduated from the Bauman Moscow Higher Technical School in 1961. Ph. D. (Eng), assoc. professor of “Optoelectronic Research Devices” department of the Bauman Moscow State Technical University. Author of about 60 scientific publications in the field of optics, optical system design.