

Островной генетический алгоритм с кроссинговерно-мутационным скрещиванием популяций для оптимальной расстановки активно-адаптивных устройств в ЭЭС большой размерности

И. С. Воронин

Санкт-Петербургский политехнический университет
Петра Великого

isvoronin@yandex.ru

Н. В. Коровкин

Санкт-Петербургский политехнический университет
Петра Великого

nikolay.korovkin@gmail.com

Аннотация. Предложен и реализован усовершенствованный островной многоцелевой генетический алгоритм (ГА). На внутреннем расчетном этапе параллельно обрабатываются независимо развивающиеся популяции-острова, на внешнем расчетном этапе выделяются наиболее перспективные острова (родители), а оставшиеся заселяются скрещиванием родителей по принципу мутации (мутируется один случайно выбранный ген во всех особях родителя) и масочного кроссинговера. Расчетные этапы повторяются в течение нескольких циклов расчета. Алгоритм позволяет решить задачу оптимальной расстановки устройств управляемой продольной компенсации (УПК) в тестовой сети IEEE из 300 узлов и получить неочевидную зависимость качества расстановки УПК от числа УПК.

Ключевые слова: островной генетический алгоритм; параллельные вычисления; многоцелевая оптимизация; оптимальная расстановка активно-адаптивных устройств; электроэнергетическая система; тестовая схема IEEE с 300 узлами

I. ВВЕДЕНИЕ

Активно адаптивные устройства (ААУ) – это силовые электронные устройства для быстрого управляемого изменения параметров электроэнергетической системы (ЭЭС) для регулирования потоков мощности, поддержания напряжения, снижения потерь, повышения статической и динамической устойчивости [1]. Практическое применение ААУ связано, в частности, с определением оптимальных мест их установки в ЭЭС. В общей постановке этой задачи ААУ характеризуется набором параметров $x \in [\underline{X}, \bar{X}]$, где \underline{X} и \bar{X} – соответственно вектора нижних и верхних пределов изменения параметров, и набором минимизируемых целевых функции $f_1(x), \dots, f_n(x)$. Для каждой возможной расстановки ААУ P необходимо построить множество Парето Π , являющееся решением задачи многоцелевой оптимизации, далее сравнить все полученные множества и найти лучшую расстановку ААУ.

При такой постановке задачи поиск решения проблематичен, а иногда, невозможен. Уже при 10 УПК

в схеме с 300 линиями электропередачи (ЛЭП) $C_{10}^{300} \sim 10^{18}$ возможных наборов мест размещения УПК. Для решения одной задачи $P \rightarrow \Pi$ необходимо выполнить $\sim 10^5$ оценок целевых функций, т.е. всего необходимо провести $\sim 10^{23}$ расчетов режима, что не реализуемо.

Для оценки целевых функций можно использовать билинейную теорему [2], что существенно (на порядки) ускорит анализ режима. Однако при учете действия на узел (ветвь) нескольких ААУ, выражения, полученные с помощью билинейной теоремы, получаются сложными. На практике одновременное регулирование режимного параметра несколькими ААУ также представляет ограниченный интерес, поскольку приводит к снижению эффективности управления, усложнению координации регуляторов и неоправданному росту стоимости системы управления. Поэтому среди расстановок ААУ более предпочтительны те, которые исключают одновременное влияние нескольких ААУ на один и тот же узел (ветвь) ЭЭС. Также более предпочтительны такие расстановки ААУ, при которых регулируется как можно больше режимных параметров. Далее два этих условия будут рассматриваться как критерии оптимальной расстановки ААУ.

В настоящей статье предлагается разбить задачу поиска оптимальных мест установки ААУ на две подзадачи:

- 1) поиск предпочтительных расстановок ААУ без анализа режимов, результатом которого является множество предпочтительных мест установки;
- 2) уточнение поиска на найденном множестве на основе расчета режимов.

Решение первой подзадачи актуально, поскольку значительно сужает пространство поиска, при котором необходим расчет режима, представляющий основную вычислительную нагрузку. При этом становится возможным решение задачи оптимальной расстановки ААУ для реальных ЭЭС с сотнями и тысячами узлов. В настоящей статье решается первая подзадача поиска предпочтительных расстановок УПК в тестовой схеме IEEE с 300 узлами.

II. КРИТЕРИИ ОПТИМАЛЬНОЙ РАССТАНОВКИ ААУ

Предложенные выше критерии оптимальной расстановки ААУ можно формализовать с помощью целевых функций $F_1(\mathbf{x})$ и $F_2(\mathbf{x})$, где вектор \mathbf{x} – конкретная расстановка ААУ (например, вектор номеров ЛЭП с УПК), а первую подзадачу представить задачей минимизации этих функций, тогда ее решение представляет собой множество Парето. Примем, что действие ААУ (УПК) с удалением от него ослабевает и регулирует режимные параметры в узлах и ветвях сети на расстоянии 1, 2 и 3 от него, а влияние на прочие узлы и ветви пренебрежимо мало. Это допущение строго не обосновано, но по нашему опыту расчетов режимов с помощью билинейной теоремы отражает отсутствие связи между режимными параметрами и ААУ. В дискуссии об этом будет сказано отдельно.

Если два ААУ одновременно воздействуют на один и тот же узел, значение целевой функции $F_1(\mathbf{x})$ увеличивается на приращение, определяемое по табл. I.

ТАБЛИЦА I. Приращения $F_1(\mathbf{x})$

Расстояние между узлом и одним ААУ	1	1	1	2	2	3
Расстояние между узлом и другим ААУ	1	2	3	2	3	3
Обозначение приращения	S_{11}	S_{12}	S_{13}	S_{22}	S_{23}	S_{33}
Величина приращения	14	10	8	6	3	1

Если на конкретный узел не воздействует ни одно ААУ, значение целевой функции $F_2(\mathbf{x})$ увеличивается на значение $S_0=1$.

Если обозначить через k_i число приращений S_i , где $i=11,12,13,22,23,33,0$, то с учетом $\delta_i=k_i(k_i+1)/2 \forall i \neq 0$ целевые функции вычисляются по формулам:

$$F_1(\mathbf{x}) = \sum_i \delta_i S_i, F_2(\mathbf{x}) = k_0 S_0.$$

Процедура определения k_i для конкретной расстановки ААУ с учетом топологии графа ЭЭС подробно рассмотрена нами в [3].

III. УСОВЕРШЕНСТВОВАННЫЙ ОСТРОВНОЙ ГА

Хотя решение первой подзадачи не требует расчета режимов, число переборов возможных мест установки ААУ велико ($\sim 10^{18}$ для схемы с 300 ЛЭП и 10 ААУ, как показано выше). Для решения задачи многоцелевой оптимизации на этом множестве предлагается усовершенствованный островной ГА. Данный алгоритм структурно разделен на внутренний и внешний расчетные этапы, которые повторяются в течение нескольких расчетных циклов. На внутреннем этапе несколько популяций-островов независимо эволюционируют с помощью параллельных вычислений. На внешнем расчетном этапе формируется новый набор популяций-островов, которые будут переданы в следующий расчетный цикл. В этот набор входят:

- наиболее перспективные острова (острова-родители), выбранные из общей популяции;

- острова, полученные скрещиванием родителей с помощью кроссинга и мутации (острова-потомки).

В отличие от классической островной модели ГА, где взаимодействие между островами осуществляется через миграцию отдельных особей [4], в предлагаемом алгоритме генетический обмен происходит на уровне целых популяций-островов, которые рассматриваются как обобщенные хромосомы.

По прошествии заданного числа расчетных циклов, острова объединяются, после выполняется сортировка недоминируемых решений, и несколько лучших фронтов Парето образуют популяцию, которая рассчитывается отдельно и выдает итоговый фронт Парето.

Алгоритм реализован в виде кода Matlab. Число УПК $N_{УПК}$, граф ЭЭС, параметры внутреннего (размер популяции N_{non} , число поколений $N_{генер}$, число островов N , доля кроссинга $\chi_{ГА}$) и внешнего (число островов-родителей $N_{род}$, доля межпопуляционного кроссинга $\chi_{скрещ}$) расчетного этапов, число расчетных циклов $N_{цикл}$ задаются при инициализации алгоритма.

IV. ВНУТРЕННИЙ РАСЧЕТНЫЙ ЭТАП

В решаемой задаче решение (особь, хромосома) \mathbf{x} – вектор номеров ЛЭП с УПК. Если число ЛЭП в схеме равно $N_{ЛЭП}$, то координаты (гены) \mathbf{x} натуральные числа от 1 до $N_{ЛЭП}$, которые должны иметь разные значения, т.к. в одну ЛЭП можно установить только одно УПК. В стандартной Matlab-реализации ГА это не предусмотрено, поэтому были прописаны оригинальные функции создания начальной популяции, кроссинга и мутации.

A. Оригинальная функция создания начальной популяции

Создается набор из $N_{non}+M$ точек (M будет определено позднее) последовательности Соболя размерностью $N_{УПК}$, далее выполняется аффинное скремблирование Матоушка-Оуэна [5, 6]. Сгенерированные точки, лежащие в гиперкубе $[0,1]^{N_{УПК}}$, умножаются на $N_{ЛЭП}-1$ и округляются до целых значений, в результате получается целочисленная начальная популяция. Каждая сгенерированная особь дополнительно проверяется и исключается, если в ней присутствуют гены с одинаковыми значениями. Вероятность p того, что в хромосоме нет повторяющихся генов, можно оценить по формуле:

$$p = \frac{A_n^k}{n^k} = \frac{n!}{n^k (n-k)!} = \prod_{i=0}^{k-1} \frac{n-i}{n}, n = N_{ЛЭП}, k = N_{УПК}.$$

Если размер начальной популяции больше N_{non} , лишние особи удаляются; если меньше N_{non} – добавляются M новых точек из последовательности Соболя, и повторяются действия, описанные выше. Значение M рассчитывается с учетом общего числа $N_{брак}$ хромосом, исключенных из начальной популяции на текущий момент:

$$\left[(N_{\text{non}} - N_{\text{брак}}) + M \right] p = N_{\text{non}} - N_{\text{брак}} \Rightarrow M = \frac{(N_{\text{non}} - N_{\text{брак}})(1-p)}{p}$$

В. Оригинальная функция кроссинга

Сначала из списка особей-родителей, подобранных Matlab на основе анализа фитнес-значений, случайно выбираются два решения $\mathbf{x}^{(1)}$ и $\mathbf{x}^{(2)}$. Далее с помощью равномерного кроссинга (uniform crossing) [7] создается особь-потомок $\mathbf{x}^{(c)}$ по схеме:

$$\mathbf{x}^{(1)}, \mathbf{x}^{(2)} \rightarrow \mathbf{x}^{(c)}: x_i^{(c)} = \begin{cases} x_i^{(1)}, & \text{если } \text{mask}_i = 1 \\ x_i^{(2)}, & \text{если } \text{mask}_i = 0 \end{cases} \quad \forall i = 1 \dots N_{\text{УПК}}$$

где mask – случайный логический вектор длиной $N_{\text{УПК}}$.

Если у особи-потомка дублируются координаты, это решение отбрасывается и создается новое. Для каждой выбранной пары особей-родителей допускается до 10 попыток сформировать корректное решение, в противном случае, пара признаётся бесперспективной, и алгоритм выбирает новых особей-родителей. Если 100 раз подряд были выбраны бесперспективные пары, то выполнение ГА останавливается, и в качестве итоговой популяции выдается популяция последнего успешно рассчитанного поколения.

С. Оригинальная функция мутации

Для проведения мутации используется вспомогательная матрица \mathbf{D} размерностью $N_{\text{ЛЭП}} \times N_{\text{ЛЭП}}$, на пересечении i -й и j -й строк которой указывается расстояние по графу между i -й и j -й ЛЭП. Матрица \mathbf{D} определяется единожды при инициализации ГА. Алгоритм мутации решения $\mathbf{x} = \{x_1, \dots, x_k, \dots, x_{N_{\text{УПК}}}\}$ приведен в табл. II.

ТАБЛИЦА II. ПРОЦЕДУРА МУТАЦИИ ОСОБЕЙ

Шаг	Действия
1	случайно равномерно выбрать мутирующий ген $k \in \{1, \dots, N_{\text{УПК}}\} \subset \mathbb{N}$
2	создать логический вектор \mathbf{m} длиной $N_{\text{ЛЭП}}$, все значения которого равны 1
3	элементам \mathbf{m} с номерами \mathbf{x} присвоить 0
4	элементам \mathbf{m} с номерами $g \in \{1, \dots, N_{\text{ЛЭП}}\} \setminus \mathbf{x}$, $g: \mathbf{D}(g, x_k) < 3 \cup \mathbf{D}(g, x_k) > 5$ присвоить 0
5	номера ненулевых элементов \mathbf{m} записать в множество Ω_k
6	случайно равномерно выбрать новое значение мутирующего гена $x'_k \in \Omega_k$
7	присвоить $\mathbf{x} = \{x_1, \dots, x'_k, \dots, x_{N_{\text{УПК}}}\}$

V. ВНЕШНИЙ РАСЧЕТНЫЙ ЭТАП

На внешнем расчетном этапе формируется пул родителей, куда добавляются первый фронт Парето общей популяции (объединение всех островов), затем последовательно острова, которые имеют хотя бы одну точку на первом, втором и т.д. фронте Парето общей популяции в порядке уменьшения числа таких точек, пока не будет достигнуто заданное число родителей.

Острова-родители для кроссинга и мутации выбираются случайно из пула родителей. Каждому j -му родителю присваивается экспоненциальный вес $w_j = a \cdot e^b \quad \forall j = 1 \dots N_{\text{род}}$, где $w_j = b = \ln(3)/(1 - N_{\text{род}})$, $a = 3/e^b$. Тогда $w_1 = 3$, $w_{N_{\text{род}}} = 1$. Затем веса нормируются до вероятностей $\xi_j = w_j / \sum w_j \quad \forall j = 1 \dots N_{\text{род}}$. Далее $\chi_{\text{скреци}}(N - N_{\text{род}})$ потомков создается кроссингом, остальные – мутацией.

А. Скрещивание по принципу кроссинга

Из пула родителей случайно с вероятностью ξ выбираются два острова-родителя $\{\mathbf{x}_1^{(1)}, \dots, \mathbf{x}_{N_{\text{non}}}^{(1)}\}$ и $\{\mathbf{x}_1^{(2)}, \dots, \mathbf{x}_{N_{\text{non}}}^{(2)}\}$, затем выполняется скрещивание:

$$\{\mathbf{x}_1^{(1)}, \dots, \mathbf{x}_{N_{\text{non}}}^{(1)}\}, \{\mathbf{x}_1^{(2)}, \dots, \mathbf{x}_{N_{\text{non}}}^{(2)}\} \rightarrow \{\mathbf{x}_1^{(c)}, \dots, \mathbf{x}_{N_{\text{non}}}^{(c)}\}: \\ \mathbf{x}_i^{(c)} = \begin{cases} \mathbf{x}_i^{(1)}, & \text{если } \text{mask}_i = 1 \\ \mathbf{x}_i^{(2)}, & \text{если } \text{mask}_i = 0 \end{cases} \quad \forall i = 1 \dots N_{\text{non}}$$

где $\{\mathbf{x}_1^{(c)}, \dots, \mathbf{x}_{N_{\text{non}}}^{(c)}\}$ – потомок, mask – случайный логический вектор длиной N_{non} .

В. Скрещивание по принципу мутации

Из пула родителей случайно с вероятностью ξ выбирается остров-родитель, затем случайно и равномерно выбирается номер мутирующего гена $k \in \{1, \dots, N_{\text{УПК}}\}$ и для каждой особи родителя выполняются шаги 2-7 процедуры мутации особей.

VI. РЕЗУЛЬТАТЫ

Представляет интерес показатель качества покрытия сети влиянием УПК и отсутствия одновременного регулирования несколькими УПК, и исследовать зависимость его величины от числа УПК. В роли этого показателя можно использовать гиперобъем [8] итогового фронта Парето из перспективных расстановок УПК. Для всех результатов приняты следующие настройки усовершенствованного островного ГА: $N_{\text{non}} = 1500$, $N_{\text{генер}} = 100$, $N = 96$, $N_{\text{род}} = 24$, $N_{\text{цикл}} = 5$, $\chi_{\text{ГА}} = 0,2$, $\chi_{\text{скреци}} = 0,4$. Гиперобъем определяется как объём (или площадь в двумерном случае) области пространства критериев, доминируемой фронтом Парето относительно заданной опорной точки. Пример вычисления гиперобъема для фронта Парето при 16 УУПК в сети IEEE с 300 узлами представлен на рис. 1.

Из-за сильно отличающихся масштабов осей гиперобъем в основном определяется улучшениями по $F_1(\mathbf{x})$, что корректируется нормировкой:

$$F_1 = \frac{F_1 - F_1^{\min}}{F_1^{\max} - F_1^{\min}}, F_2 = \frac{F_2 - F_2^{\min}}{F_2^{\max} - F_2^{\min}}$$

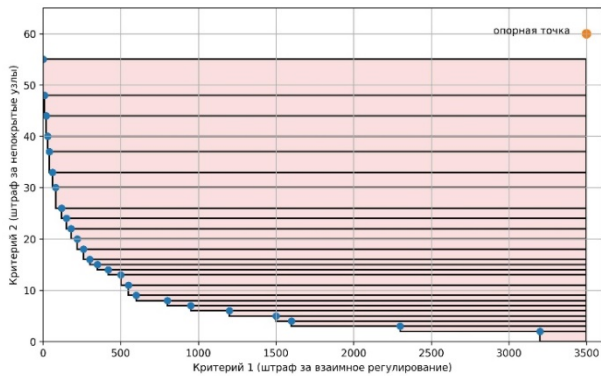


Рис. 1. Пример вычисления гиперобъема

Тогда обе цели будут лежать в диапазоне $[0,1]$ и в равной степени влиять на величину гиперобъема. С учетом фактических значений целевых функций при изменении числа УПК от 8 до 24, для любого числа УПК принимается $F_1^{\min} = F_2^{\min} = 0$, $F_1^{\max} = 16000$, $F_2^{\max} = 75$, опорная точка после нормировки (1,1).

Полученные значения гиперобъема HV представлены в табл. III и на рис. 2.

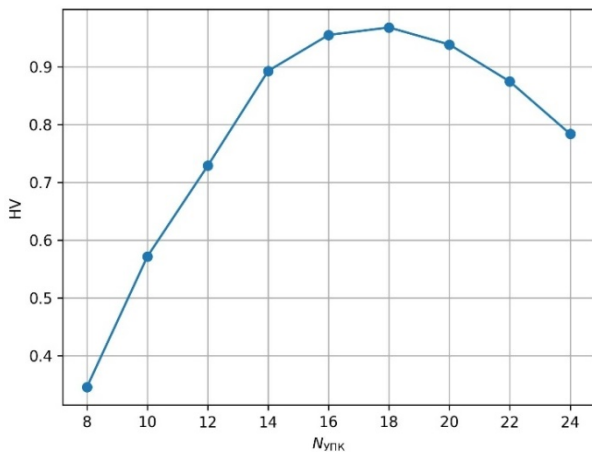


Рис. 2. Зависимость гиперобъема от числа УПК

ТАБЛИЦА III. ЗНАЧЕНИЯ ГИПЕРОБЪЕМА ПРИ РАЗНОМ ЧИСЛЕ УПК

$N_{уПК}$	HV	$N_{уПК}$	HV	$N_{уПК}$	HV
8	0,3455	14	0,8927	20	0,9383
10	0,5714	16	0,9549	22	0,8743
12	0,7285	18	0,9679	24	0,7835

Из результатов видно, что значение гиперобъема возрастает с увеличением числа УПК, достигая максимального значения $HV_{\max} = 0,9679$ при 18 УПК, затем уменьшается. Это можно объяснить следующим образом. При увеличении числа УПК от 8 до 18 становится меньше непокрытых узлов при относительно небольшом увеличении штрафа за одновременное регулирование (в сети есть место для новых УПК, все устройства работают максимально эффективно, не мешая друг другу). При увеличении числа УПК от 18 до 24 достигается критическая плотность размещения, при которой новые устройства практически не уменьшают штраф за непокрытые влиянием узлы, при этом все больше УПК одновременно влияют на одни и те же элементы сети.

VII. ДИСКУССИЯ

Допущение об уменьшении влияния ААУ при увеличении расстояния по графу требует уточнения. В дальнейшем можно для каждой ЛЭП в графе ЭЭС ввести вес, например, пропускную способность, и при расчете целевой функции $F_1(x)$ использовать взвешенные расстояния по графу.

Перенос предложенного механизма генетического обмена на алгоритм MAP-Elites [9] представляет перспективное направление дальнейших исследований. Каждая ячейка карты может рассматриваться не как отдельная особь, как популяция решений. Это позволит применять операторы кроссинга и мутации на уровне популяций, обеспечивая межнишевую рекомбинацию и направленное исследование пространства дескрипторов.

VIII. ЗАКЛЮЧЕНИЕ

В настоящей статье были предложены:

1) подход к решению задачи оптимальной расстановки ААУ в ЭЭС большой размерности (сотни и больше узлов)

2) усовершенствованный островной ГА, в котором вместо миграции отдельных особей выполняется скрещивание популяций-островов по тем же принципам, по которым обычный ГА скрещивает особи в рамках отбора. Аналогичный подход можно применить в MAP-Elites, что представляется перспективным.

3) подход, позволяющий оценить качество полученных алгоритмом решений и выбрать оптимальное число УПК для рассматриваемой сети. Для сети IEEE-300 расстановка УПК наиболее эффективна при 18 устройствах.

СПИСОК ЛИТЕРАТУРЫ

- [1] Hingorani N.G., Gyugyi L. Understanding FACTS: Concepts and Technology of Flexible AC Transmission Systems. New York: IEEE Press, 2000. 432 p.
- [2] Беляев Н.А., Коровкин Н.В., Фролов О.В., Чудный В.С. Использование билинейной теоремы для решения задач оптимизации потоков мощностей в энергосистемах // Электротехнические комплексы и системы управления. 2012. № 1. С. 77–80.
- [3] Воронин И.С., Коровкин Н.В., Долбин К.С. Оптимальная расстановка активно-адаптивных устройств для моделей энергосистем большой размерности // Глобальная энергия. 2025. Т. 31. № 4. С. 26–40. DOI: 10.18721/JEST.31402.
- [4] Whitley D., Rana S., Heckendorn R.B. The Island Model Genetic Algorithm: On Separability, Population Size and Convergence // Journal of Computing and Information Technology. 1999. Vol. 7. No. 1. P. 33–47.
- [5] Owen A.B. Randomly Permuted (t, m, s)-Nets and (t, s)-Sequences // Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing. New York: Springer, 1995. P. 299–317. DOI: 10.1007/978-1-4612-2552-2_19.
- [6] Matoušek J. On the L2-Discrepancy for Anchored Boxes // Journal of Complexity. 1998. Vol. 14, No. 4. P. 527–556. DOI: 10.1006/jcom.1998.0489.
- [7] Syswerda G. Uniform Crossover in Genetic Algorithms // Proceedings of the 3rd International Conference on Genetic Algorithms. San Mateo, CA: Morgan Kaufmann, 1989. P. 2–9.
- [8] Zitzler E., Thiele L. Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach // IEEE Transactions on Evolutionary Computation. 1999. Vol. 3. No. 4. P.257–271. DOI: 10.1109/4235.797969.
- [9] Mouret J.-B., Clune J. Illuminating Search Spaces by Mapping Elites // arXiv preprint. 2015. arXiv:1504.04909.