

АЛГОРИТМ ДИФФЕРЕНЦИАЛЬНОЙ ЭВОЛЮЦИИ ДЛЯ ОПТИМИЗАЦИИ НАПРАВЛЕННОСТИ ФАЗИРОВАННЫХ АНТЕННЫХ РЕШЕТОК

А.В. Еремеев¹

доцент, д.ф.-м.н., e-mail: eremeev@ofim.oscsbras.ru

Н.Н. Тюнин¹

аспирант, e-mail: n.n.tyunin@gmail.com

¹Институт математики им. С.Л. Соболева СО РАН, Омский филиал, Омск

Аннотация. Задача оптимизации направленности фазированных антенных решеток для радиосвязи коротковолнового диапазона сформулирована как невыпуклая задача квадратичного программирования. Метод дифференциальной эволюции адаптирован к специфике рассматриваемой задачи и проведен вычислительный эксперимент, в котором предложенный алгоритм сравнивается с пакетом BARON, основанном на многократном применении алгоритма локального поиска и методе ветвей и границ.

Ключевые слова: Задача оптимизации, фазированная антенная решетка, квадратичное программирование, коротковолновый диапазон, дифференциальная эволюция.

Введение

В настоящее время одной из актуальных задач в области разработки и анализа эффективных систем радиосвязи является задача оптимизации направленности фазированных антенных решеток (ФАР), представляющих собой антенные системы, распределение фаз и амплитуд на элементах которых позволяет получать направленное излучение. Будучи собранными в антенную систему и разведенными в пространстве, излучатели формируют диаграмму направленности, которая зависит от расположения и конструкции излучателей, а также выбора фаз и амплитуд сигналов, подаваемых на вход излучателей.

В диапазоне сверхвысоких частот (СВЧ) задачи оптимизации фаз и амплитуд излучателей, как правило, решаются с использованием некоторых упрощающих предположений [1–3]. Однако, в диапазоне высоких частот (ВЧ) задача оптимизации направленности ФАР оказывается более сложной, и потому менее изучена [4, 5]. При ограничении суммарной мощности, подаваемой на антенную систему, задача выбора фаз и амплитуд на излучателях может быть решена аналитически [6]. Однако, при ограничении на мощность по каждому входу антенной системы требуется решение невыпуклых задач квадратичного программирования [7]. Вообще говоря, задачи квадратичного программирования являются NP-трудными [8]. Для решения таких задач могут применяться

методы ветвей и границ [9], отсечений [10], ДС-программирования [11], полуопределенной релаксации [7], эволюционных вычислений [12, 13], локального поиска [14] и др.

Настоящая работа посвящена разработке алгоритмов решения задачи оптимизации направленного излучения ФАР ВЧ диапазона с использованием градиентного подъема и эволюционных вычислений. Результаты данной работы были кратко представлены в [15].

1. Основные обозначения

Математическая постановка задачи оптимизации направленности ФАР формулируется из физических соображений в виде выпуклого квадратичного функционала при невыпуклых квадратичных ограничениях [6]:

$$\begin{cases} u^+ A u \rightarrow \max, \\ 0 \leq u^+ B^{(1)} u \leq, \\ \dots \\ 0 \leq u^+ B^{(N)} u \leq 1. \\ u \in \mathbb{C}^N \end{cases} \quad (1)$$

Здесь n – количество излучателей в системе. Матрица A описывает влияние каждого элемента системы на ее излучение вдоль заданного направления. Ограничения, описываемые матрицами $B_k, k = \overline{1, N}$, накладываются на мощность, которую генератор сигнала способен подать на каждый излучатель. Вектор u представляет собой вектор-столбец комплексных напряжений.

Для построения алгоритмов анализа и решения данной задачи был предложен переход от постановки в комплексных числах к постановке в вещественных [16]:

$$\begin{cases} x^T G x \rightarrow \max, \\ 0 \leq x^T H^{(1)} x \leq 1, \\ \dots \\ 0 \leq x^T H^{(N)} x \leq 1, \\ x \in \mathbb{R}^{2n}. \end{cases} \quad (2)$$

Здесь матрица G вычисляется по матрице A , а матрицы $H_k, k = \overline{1, N}$ – по матрицам $B_k, k = \overline{1, N}$.

Ранее было показано [16], что задача (1) имеет тривиальную симметрию $u \rightarrow e^{j\phi} u$, где j – мнимая единица, ϕ – фазовый сдвиг, одинаковый во всех компонентах вектора u . При исследовании структуры локальных оптимумов в указанной работе было отмечено, что, возможно, задача (1) имеет и другие симметрии.

Также, в работе [16] для отыскания локального оптимума использовался многократный запуск градиентного подъема. Здесь для этих целей предлагает-

ся гибридный алгоритм, сочетающий элементы градиентного метода и дифференциальной эволюции.

При этом, задача (2) сводится к задаче безусловной оптимизации методом внешней точки в виде (3)

$$x^T G x - r \cdot \sum_{k=1}^n \left(\min \left(0, x^T H^{(k)} x \right) + \min \left(0, 1 - x^T H^{(k)} x \right) \right)^a \rightarrow \max, \quad (3)$$

где r и a – настраиваемые параметры.

2. Дифференциальная эволюция

2.1. Базовый вариант алгоритма

Эволюционные алгоритмы (ЭА) — один из наиболее широко используемых методов решения многоэкстремальных задач оптимизации, применяемый во многих областях информатики, экономики, инженерии и др. Дифференциальная эволюция (ДЭ) [17] — один из наиболее эффективных ЭА непрерывной оптимизации. В частности, ДЭ была признана стратегией-победителем нескольких конкурсов по оптимизации [18]. Подобно другим эволюционным алгоритмам, ДЭ вдохновлен естественным процессом эволюции и включает в себя применение мутаций, рекомбинации и селекции. Основная особенность ДЭ заключается в том, что в этом алгоритме при построении особей-потомков учитываются разницы между векторами, присутствующими в популяции. В этом смысле он похож на алгоритм Нелдера-Мида [19] и метод CRS из [20].

ДЭ – рандомизированный алгоритм, основанный на популяционном поиске, в котором на каждой итерации вычисляется новый набор пробных решений (векторов). В базовом варианте ДЭ для каждого члена популяции (их называют целевыми векторами) создается новый мутантный вектор. Затем мутантный вектор комбинируют с целевым вектором для создания пробного вектора. Наконец, применяется фаза селекции для выбора особей следующей популяции. Таким образом, итерации ДЭ продолжаются, пока не будет достигнут критерий остановки. В поколении G i -й вектор популяции обозначается как $X_{i,G} = [x_{1,i,G}, x_{2,i,G}, \dots, x_{D,i,G}]$. Ниже приведены более подробные сведения о каждой фазе ДЭ.

Эксперименты показывают [21], что в целом эволюция популяции такова, что в случае попадания популяции в некоторый «овраг», популяция ориентируется вдоль этого оврага и математическое ожидание разности двух случайных векторов оказывается направленным вдоль оврага. Это обеспечивает быстрое движение вдоль узких оврагов, тогда как для градиентных методов в аналогичных условиях характерна колебательная динамика «от стенки к стенке». Приведенные эвристические соображения иллюстрируют наиболее важную и привлекательную особенность алгоритма ДЭ – способность динамически адаптироваться к особенностям рельефа оптимизируемой функции. Именно этим объясняется способность алгоритма быстро проходить сложные овраги, обеспечивающая его способность к поиску оптимальных или близких к оптимуму решений.

Инициализация. ДЭ обычно начинает процесс оптимизации со случайно иницизированной популяции, состоящей из D векторов. Поскольку информация о перспективности различных областей пространства решений, как правило, отсутствует, для инициализации ДЭ применяются равномерные генераторы случайных векторов в параллелепипеде с некоторыми границами a_i , b_i по каждой координате j , т.е. j -я компонента i -го вектора инициализируется как $x_{j,i,0} = a_j + rand_{i,j}(b_j - a_j)$, где $rand_{i,j}$ – равномерно распределенное случайное число, лежащее между 0 и 1.

Мутация. Для каждого целевого вектора создается мутантный вектор. В настоящее время известно несколько вариантов этого оператора. В классическом варианте ДЭ применяется стратегия $rand/1$. В этом случае мутантный вектор $V_{i,G}$ создается следующим образом:

$$V_{i,G} = X_{r1,G} + F \times (X_{r2,G} - X_{r3,G}), \quad r_1 \neq r_2 \neq r_3, \quad (4)$$

где r_1, r_2, r_3 – попарно различные целые числа, случайно выбранные из множества $\{1, \dots, D\} \setminus i$. Разница между векторами масштабируется множителем F , который называется силой мутации и обычно определяется в интервале $[0.4, 1]$.

Рекомбинация. Для объединения информации о различных пробных решениях и с целью увеличения разнообразия в популяции применяется оператор кроссинговера (рекомбинации). Каждый целевой вектор $X_{i,G}$ рекомбинируется с соответствующим ему мутантным вектором $V_{i,G}$ для создания пробного решения $U_{i,G} = [u_{1,i,G}, u_{2,i,G}, \dots, u_{D,i,G}]$. Наиболее типичный вариант кроссовера в ДЭ действует следующим образом:

$$u_{j,i,G} = \begin{cases} v_{j,i,G}, & \text{если } rand_{i,j} \leq CR \text{ или } j = j_{rand}, \\ x_{j,i,G} & \text{иначе,} \end{cases} \quad (5)$$

где $rand_{i,j}$ – равномерно на $[0, 1]$ распределенное случайное число, j_{rand} – случайно выбранный индекс. При этом гарантируется, что $U_{i,G}$ наследует хотя бы одну компоненту от $V_{i,G}$. Параметр $CR \in [0, 1]$ называется интенсивностью кроссинговера.

Селекция. Наконец, выполняется отбор для определения следующего поколения. Каждый пробный вектор сравнивается с соответствующим ему целевым вектором, и выживает лучший из них:

$$X_{i,G+1} = \begin{cases} U_{i,G}, & \text{если } \tilde{F}(U_{i,G}) \leq \tilde{F}(X_{i,G}) \\ X_{i,G} & \text{иначе.} \end{cases} \quad (6)$$

Таким образом, каждый член популяции либо становится лучше по целевой функции, либо остается с тем же значением целевой функции в следующем поколении.

2.2. Гибридный вариант алгоритма

Будучи примененным к различным примерам из [16], описанный выше базовый вариант алгоритма дифференциальной эволюции зачастую не мог обнаружить даже допустимых решений за все отведенное ему время. Чтобы избежать этой тенденции, в данной работе предложен гибридный вариант ДЭ с использованием градиентного метода, масштабированием решений в допустимую область и адаптацией штрафа, учитывающий специфику решаемой задачи.

Комбинация ДЭ с градиентным подъемом реализуется следующей процедурой, выполняемой перед переходом к следующему поколению $G + 1$:

Пусть G_0 – номер поколения, когда было последнее улучшение рекорда,

D – размер популяции,

$Grad(X)$ – результат применения градиентного подъема с начальным решением X ,

X_G – лучшая особь популяции на итерации G .

Если $G > D$ и $i > 2G_0$ то положить

$X' := Grad(X_G)$,

Если $X' = X_G$, то завершить выполнение алгоритма, выдать решение X_G .

Иначе положить

$X_G := X'$,

$G_0 := G$.

Условие $G > 2G_0$ определяет, происходило ли улучшение популяции за определенное количество итераций. Условие $G > D$ гарантирует, что на начальных итерациях ДЭ не будет принято решение о запуске градиентного алгоритма [22].

Масштабирование решений. Для рассматриваемой задачи существует преобразование, позволяющее привести к допустимой области любое решение x , которое нарушает только неравенства задачи (3) вида $x^T H^{(k)} x \leq 1$:

$$x' := \alpha(x)^{-1/2} x, \quad (7)$$

где $\alpha(x) := \max_{k=\overline{1,n}} x^T H^{(k)} x$. Поскольку как целевая функция, так и ограничения представлены квадратичными формами, применение такой операции приведет к пропорциональному уменьшению в $\alpha(x)$ раз значений каждой из квадратичных форм. Другими словами, если в некоторой точке x значения каждой из квадратичных форм, задающих ограничения, больше 0, причем значения некоторых из них больше 1, то по формуле (7) можно определить множитель, умножение которого на вектор x ведет к тому, что наибольшее из значений квадратичных форм, задающих ограничения, будет равно 1. Данная процедура применяется для выбора начального решения, а также для масштабирования каждого нового решения перед оценкой его качества в алгоритме ДЭ. При этом если решение до масштабирования имело большие нарушения ограничений, то после масштабирования его целевая функция существенно изменяется (как правило, в сторону снижения качества). На этом принципе основано адаптивное правило подбора штрафного коэффициента r , описываемое далее.

Пусть

G – номер текущей итерации,

G_0 – номер итерации, на которой было получено улучшение рекорда,

G_1 – номер итерации, на которой произошло предыдущее увеличение штрафа,

D – размер популяции,

r – текущее значение штрафного коэффициента.

Если $G > D$ и $G > 1.5G_0$ и $G > 2G_1$ то положить

$$r := 2r$$

$$G_1 := G.$$

Здесь условие $G > D$ гарантирует, что на начальных итерациях не будет принято решение об увеличении штрафа, а условие $G > 1.5G_0$ – что лучшее значение популяции не было улучшено за определенное количество итераций. Условие $G > 2G_1$ откладывает следующее срабатывание критерия, чтобы штрафной коэффициент не увеличивался слишком быстро. Таким образом, при отсутствии существенных улучшений качества решений происходит увеличение штрафного коэффициента. С учетом масштабирования особей в допустимую область, это приводит к выживанию особей с меньшим нарушением ограничений и позволяет улучшить качество получаемых решений.

2.3. Вычислительный эксперимент

Вычислительный эксперимент был поставлен для задач, рассмотренных в [16, 23]. В данной статье ШВИК, ШВДК, СВДК обозначают решетки кольцевой структуры. Далее следует указание количества элементов (8 или 16). Через тире – расстояние от центра излучателя до центра решетки (15, 20, 25, 30, 37). Для ШВД приводится плотность противовесов (например, 3:3). Поскольку современные ЭВМ на аппаратном уровне поддерживают вычисление в параллельном режиме, рассматриваемый здесь алгоритм ДЭ был адаптирован под эту особенность: за один запуск алгоритма производится 4 параллельных выполнения, а на выход подается лучшее решение. Это позволяет использовать возможности современных ЭВМ для получения более качественных решений. В таблице 1 результаты, полученные с помощью ДЭ, сравниваются с результатами коммерческого решателя BARON и в пакете GAMS. Решатель BARON, также как ДЭ, имеют ограничение по времени счета 1000с (эта длительность выбрана из практических соображений, т.к. она сопоставима с временем построения исходных данных с использованием системы NEC, и совпадает с ограничением по времени, выбранным в [16]). В случае нескольких запусков ДЭ за отведенное время, за решение принимается лучшее из найденных за 1000с. Для оценки среднего значения целевой функции на выходе ДЭ производилась серия из 10 независимых испытаний по 1000с в каждом. Результаты эксперимента при выборе типичных значений настраиваемых параметров ДЭ (популяция из 100 особей, $F = 0.6$, $CR = 0.6$) представлены в таблице 1. Значения целевой функции в таблице округлены до целых. Полужирным шрифтом выделены случаи, когда указанное значение целевой функции не менее, чем на 1% выше, чем у другого алгоритма. Вычисления производились на ЭВМ с процессором Intel i7

(тактыовая частота: 2.8ГГц), ОЗУ: 16Гб.

Таблица 1. Результаты экспериментов, полученные с помощью ДЭ и BARON

Тип	ДЭ \tilde{F}	BARON \tilde{F}	t, c
ШВИ 2x2	139	139	0.12
ШВИ 3x3	580	580	0.34
ШВД 2x2	463	463	0.27
ШВД 3x3	924	925	0.34
СВД 2x2	361	361	0.16
СВД 3x3	1163	1261	0.38
СВД 5x5	7132	6716	1000
СВД' 2x2	198	253	0.25
СВД' 3x3	834	1153	1.4
СВД' 5x5	2755	33	217.94
ШВИК 8-15(3:3)	218	218	0.23
ШВИК 16-15(3:7)	732	734	1.37
ШВИК 8-15(2:3)	1664	-	14.62
ШВДК 8-20	1454	1454	2.78
ШВДК 8-30	2421	2421	1.47
СВДК 8-25	732	734	0.23
СВДК 8-37	1486	1486	0.23

Аналогичные испытания производились и с помощью решателя ANTIGONE в пакете GAMS, однако, было выявлено, что в режиме по умолчанию данный решатель на всех задачах выдал нулевое решение, кроме СВД 2x2, где решение по целевой функции совпадало с результатом BARON. Также, в результате проведенных экспериментов было обнаружено, что динамическая адаптация штрафного коэффициента позволяет ДЭ достичь более качественных решений, что особенно заметно в тех случаях, когда результаты ДЭ и BARON близки по качеству.

С целью изучения возможности ускорения работы решателей за счет учета специфики задачи были проведены дополнительные исследования структуры рассматриваемых примеров с точки зрения линейных симметрий этих задач [24]. Ранее было отмечено (см. [16]), что решения рассматриваемой задачи эквивалентны с точностью до сдвига фаз во всех излучателях на равную величину. Учет данной симметрии (для краткости называемой «фазовой симметрией») может быть реализован фиксацией в ноль одной из переменных задачи, например, $x_1 = 0$. В результате добавления такого ограничения к условиям задачи число переменных сокращается на единицу и можно предположить, что это сократит время вычислений для известных алгоритмов. Нами поставлены вопросы о том, действительно ли происходит сокращение времени вычислений и о существовании других семейств симметрий, которые могли бы еще более сократить пространство поиска решаемой задачи.

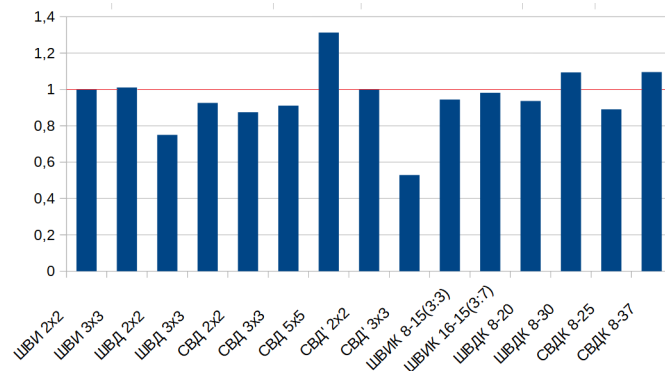


Рис. 1. Отношение длительности вычислений с фиксацией первой координаты к исходной длительности вычислений

Для ответа на первый вопрос на всех тестовых примерах был найден коэффициент ускорения решателя BARON, получаемый от фиксации $x_1 = 0$. Результаты представлены на рисунке 1 (здесь отсутствует задача СВД' 5x5, где алгоритм с фиксацией затратил существенно большее время, но при этом нашел решение с большим на 10% значением целевой функции, а также задача ШВИК 8-15(2:3), в которой решения не были найдены в обоих случаях). большинстве примеров фиксация переменной привела к ускорению работы алгоритма, среднее ускорение по представленным здесь задачам составило 0.95, что говорит о целесообразности фиксации в ноль одной из переменных при использовании решателя. Аналогичный эксперимент с алгоритмом ДЭ и решателем ANTIGONE не показал существенного улучшения качества решений или их ускорения в результате фиксации x_1 .

Для ответа на второй вопрос был проведен поиск непрерывной группы линейных симметрий в рассматриваемых примерах, как описано в [24]. В результате было установлено, что других непрерывных семейств линейных симметрий в рассматриваемых примерах не существует. Вопрос о поиске дискретных симметрий пока остается открытым.

Из проведенных экспериментов можно сделать следующие выводы:

- Разработанный в рамках данной работы гибридный вариант ДЭ показывает конкурентоспособные результаты по сравнению с коммерческим решателем BARON в режиме его настроек по умолчанию, при этом преимущество ДЭ наблюдается на задачах с наибольшей размерностью (25 и 16 переменных).
- Решатель ANTIGONE в режиме его настроек по умолчанию в большинстве тестовых примеров возвращает нулевое решение.
- Учет фазовых симметрий задачи (1), как правило, позволяет ускорить работу решателя BARON.
- Других непрерывных семейств линейных симметрий в рассматриваемых задачах не существует.

3. Заключение

В рамках данной работы был разработан гибридный вариант алгоритма дифференциальной эволюции с использованием градиентного алгоритма и адаптацией штрафа. Показано, что разработанный вариант ДЭ демонстрирует конкурентоспособные результаты.

В целях ускорения работы алгоритмов для рассмотренных задач была применена методика анализа наличия группы непрерывных симметрий, в результате которой было обнаружено наличие только фазовой симметрии. Было произведено сравнение гибридного алгоритма и коммерческих решателей с учетом обнаруженной симметрии и без нее. Выявлено, что учет симметрии может привести к ускорению работы решателя BARON.

Благодарности

Авторы благодарны А.С. Юркову за постановку задачи и полезные замечания. Работа выполнена в рамках государственного задания ИМ СО РАН (проект FWNF-2022-0020).

Литература

1. Indenbom M., Izhutkin V., Sharapov A., Zonov A. Synthesis of Conical Phased Antenna Arrays Optimization of Amplitude Distribution Parameters // IX International Conference on Optimization and Applications (OPTIMA 2018). 2018. P. 273–285.
2. Шелкунов С. А., Фрис Г. Антенны: Теория и практика. Советское радио, Москва, 1955.
3. Фаняев И. А., Кудин В. П. Синтез амплитудного распределения на входах излучателей фазированной антенной решетки над цилиндрической поверхностью методом роя частиц // Доклады БГУИР. 2017. С. 89–96.
4. Фаняев И. А., Кудин В. П. Фазированная антенная решетка кругового обзора над проводящей цилиндрической поверхностью из излучателей вертикальной поляризации // Известия Гомельского государственного университета имени Ф. Скорины. Сер.: Естественные науки. 2014. С. 191–198.
5. Юрков А. С. Максимизация направленности фазированных антенных решеток коротковолнового диапазона // Техника радиосвязи. ОНИИП, Омск, 2016. № 2, С. 46–53.
6. Юрков А. С. Оптимизация возбуждения передающих фазированных антенных решеток декаметрового диапазона длин волн. ОНИИП, Омск, 2014.
7. Fuchs B. Application of convex relaxation to array synthesis problems // IEEE Transactions on Antennas and Propagation 62(2). 2014. P. 634–640.
8. Murty K. Some NP-complete Problems in Quadratic and Nonlinear Programming // Mathematical Programming. North Holland 39. 1987. P. 117–129.
9. Нечаева М. С., Хамисов О. В. Метод ветвей и границ для задачи минимизации невыпуклой квадратичной функции при выпуклых квадратичных ограничениях // Дискретн. анализ и исслед. опер. 7(2). 2000. С. 74–88.

10. Horst R., Pardalos P. M. Handbook of global optimization. Springer Science & Business Media, 2013. Vol. 2.
11. Стрекаловский А. С. О минимизации разности выпуклых функций на допустимом множестве //Ж. вычисл. матем. и матем. физ. 43(3). 2003. С. 399–409.
12. Boriskin A. V., Balaban M. V., Galan O. Yu., Sauleau R. Efficient approach for fast synthesis of phased arrays with the aid of a hybrid genetic algorithm and a smart feed representation //2010 IEEE International Symposium on Phased Array Systems and Technology. 2010. P. 827–832.
13. Rao A., Sarma N. Synthesis of reconfigurable antenna array using differential evolution algorithm //IETE Journal of Research 63(3). 2017. P. 827–832.
14. Кочетов Ю. А. Вычислительные возможности локального поиска в комбинаторной оптимизации //Ж. вычисл. матем. и матем. физ. 48(5). 2008. С. 747–763.
15. Eremeev A., Tiunin N. Differential evolution for short wave antenna array optimization //Mathematical Optimization Theory and Operations Research. 2022. <https://easychair.org/smart-program/MOTOR2022/2022-07-04.html#talk:195757> (accessed: 27.08.2022).
16. Тюнин Н. Н. Задачи невыпуклого квадратичного программирования, связанные с оптимизацией фазированных антенных решёток //ДАИО 28(3). 2021. С. 65–89.
17. Storn R., Price K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces // Journal of global optimization. Springer, 1997. Vol. 11, № 4, P. 341–359.
18. Das S., Suganthan P. N. Differential evolution: A survey of the state-of-the-art //IEEE transactions on evolutionary computation 15(1). 2011.
19. Nelder J. A., Mead R. A simplex method for function minimization //The computer journal 7(4). 1965. P.308–311.
20. Price W. Journal of Optimization Theory and Applications //The computer journal 40(3). 1983. P.333–348.
21. Price K. V., Storn R. M., Lampinen J. A. Differential Evolution. A Practical Approach to Global Optimization. Springer, 2005.
22. Eremeev A. V. A Restarting Rule Based on the Schnabel Census for Genetic Algorithms //Battiti, R., Brunato, M., Kotsireas, I., Pardalos, P. (eds) Learning and Intelligent Optimization. LION 12 2018. Lecture Notes in Computer Science, Springer 11353. 2018.
23. Тюнин Н. Н. Об оптимизации направленности коротковолновых фазированных антенных решеток кольцевой структуры //Техника радиосвязи, ОННП. 2022.
24. Eremeev A. V., Yurkov A. S. On Symmetry Groups of Some Quadratic Programming Problems //Mathematical Optimization Theory and Operations Research. Proc. 19th Int. Conf. MOTOR 2020 (Novosibirsk, Russia, July 6–10, 2020). Springer, LNCS 12095. 2020. P.35–48.

A.V. Ereemeev¹

Dr. Sci. (Phys.-Math.), e-mail: eremeev@ofim.oscsbras.ru

N.N. Tyunin²

p.g., e-mail: n.n.tyunin@gmail.com

¹Dostoevsky Omsk State University, Omsk

²Sobolev Institute of Mathematics, SB RAS, Omsk department, Omsk

Abstract. The problem of the directivity optimization of phased antenna arrays for shortwave radio communication is formulated as a non-convex quadratic programming problem. The differential evolution method is adapted to the specifics of the problem under consideration and a computational experiment was carried out in which the proposed algorithm is compared with the BARON package, based on repeated application of the local search algorithm and branch and bound method.

Keywords: Optimization problem, phased antenna array, quadratic programming, shortwave, differential evolution.