

С. П. Симаков, аспирант, e-mail: simbal393@gmail.com,  
Самарский университет имени академика С. П. Королева  
(национальный исследовательский университет)

### Статистическое исследование генетического алгоритма дифференциальной эволюции для обработки оптических изображений

*Рассмотрено использование генетического алгоритма дифференциальной эволюции для определения ориентации и положения объекта относительно оптического сенсора по серии кадров, сделанных сенсором. В качестве измерительных средств при этом используются коммерческие оптические камеры. Проведено статистическое исследование генетического алгоритма дифференциальной эволюции на модельной задаче отделения наноспутника от орбитальной ступени, включающее в себя исследование влияния параметров настройки алгоритма (гиперпараметров) на точность получаемого результата. Валидация оценок проводится с использованием нелинейного метода наименьших квадратов. Приведены зависимости ошибок в определении относительной ориентации и относительного положения, полученные с помощью двух алгоритмов. Исследование показывает, что генетический алгоритм дифференциальной эволюции можно применять в различных задачах обработки информации с оптических датчиков.*

**Ключевые слова:** наноспутник, метод наименьших квадратов, оптический сенсор, относительные параметры движения, генетический алгоритм дифференциальной эволюции, инспектор, объект наблюдения

#### Введение

Нарастающие усложнение и увеличение информационного потока, а также увеличение скорости обработки данных позволяют решать задачи, считавшиеся раньше крайне энергозатратными и ресурсоемкими. Одной из таких является задача обработки измерительной информации, получаемой с оптического сенсора — кино/фотокамеры, для задач идентификации быстропротекающих процессов в космическом пространстве, таких как разделение космических аппаратов или их частей, раскрытие элементов конструкции.

При этом под идентификацией быстропротекающих процессов в механических системах в данной работе понимается определение характеристик как движения центра масс объекта, так и движения объекта относительно центра масс.

Рассмотрим методы обработки информации, полученной посредством оптического датчика, более подробно.

Одна из возможных методик определения параметров движения объекта относительно датчика представлена в работе [1]. Новый подход,

содержащий сигма-точечный фильтр Калмана, используется здесь для определения параметров движения. Также в работе используется особая модель измерений, которая преобразует полученные по результатам обработки видеоизображения пиксельные координаты ориентиров в компоненты линии визирования [2, 3].

В статье [4] рассматривается экспериментальное определение линейных и угловых скоростей вращающегося цилиндра двумя экспериментальными методами, один из которых является оптическим. Для фиксирования положения цилиндра при этом применяется скоростная видеокамера.

В работе [5] описывается подход, в основу которого положено использование информации, получаемой с оптического датчика. Авторы проводят с помощью разработанного подхода верификацию эксперимента по сближению двух инженерных моделей космических аппаратов. Но данный метод проверяется на работоспособность только в плоском случае, что является несомненным минусом работы. Таким образом, встает вопрос о разработке подхода, способного определять положение и ориентацию объекта в пространстве с заданной точностью.

Целью данной работы является статистическое исследование генетического алгоритма дифференциальной эволюции (DE) в его применении к задачам обработки оптических изображений для определения характеристик углового и линейного движения инспектора относительного объекта наблюдения, валидация данного алгоритма нелинейным методом наименьших квадратов (NLS).

Этап обработки и слежения за нужными сегментами изображений в данной работе не рассматривается. Предполагается, что положение реперных точек на поверхности объекта наблюдения относительно его центра масс известно заранее.

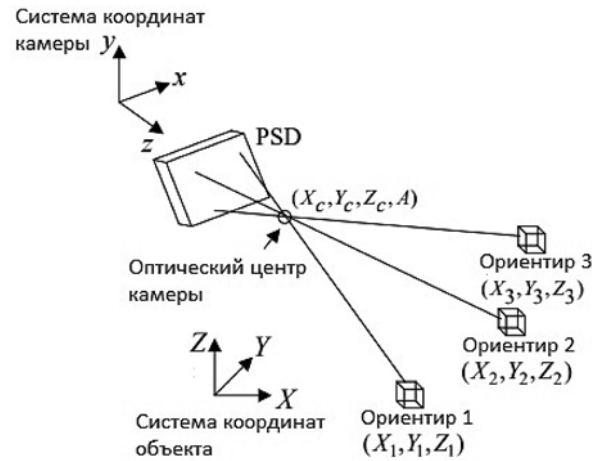


Рис. 1. Выбранные системы координат

### Постановка задачи и математическая модель оптического сенсора

Будем использовать следующие системы координат:

1. Связанная система координат объекта наблюдения (ССК)  $OXYZ$  с началом в центре масс объекта. Ось  $OX$  направлена вдоль продольной оси объекта. Ось  $OY$  располагается в плоскости симметрии, совпадающей с плоскостью траектории, и направлена вверх. Ось  $OZ$  дополняет систему координат до правой.

2. Система координат, связанная с оптическим сенсором (СКС), установленном на инспекторе,  $O_x y z$  с началом в оптическом центре камеры. Ось  $z$  направлена параллельно оптической оси. Ось  $x$  и  $y$  формируют проецирующую плоскость  $O_x y$ .

3. Система координат плоскости матрицы оптического сенсора (СКК)  $O_p uv$  с началом в верхнем левом углу фотокадра. Ось  $O_p u$  направлена по горизонтали вправо, ось  $O_p v$  — по вертикали вниз.

Выбранные системы координат представлены на рис. 1, 2.

Таким образом, под понятием относительного положения понимается положение центра масс объекта наблюдения относительно сенсора, установленного на инспекторе, в системе координат, связанной с сенсором.

В данной работе также используются модифицированные параметры Родриго (MRP) [6] для описания вращательного движения.

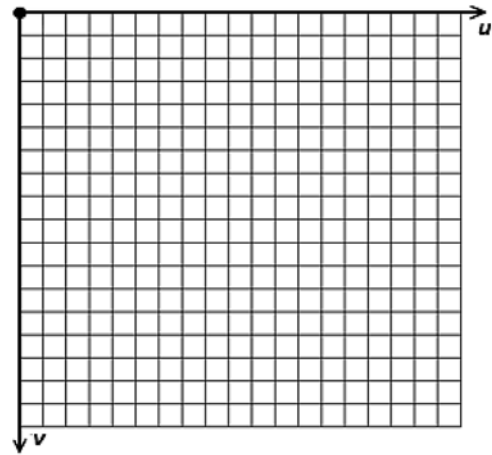


Рис. 2. Система координат плоскости матрицы оптического сенсора

MRP можно определить как

$$\sigma = [\sigma_1, \sigma_2, \sigma_3]^T = \text{etg} \left( \frac{\Phi}{4} \right),$$

где  $\mathbf{e} = [e_1, e_2, e_3]^T$  — единичный вектор, направление которого совпадает с осью вращения;  $\Phi$  — угол вращения.

Матрица перехода между системами координат имеет вид

$$A = [I_{3 \times 3}] \frac{8\sigma^2 - 4(1 - \sigma^2)\sigma}{(1 + \sigma^2)^2},$$

или в развернутой форме:

$$A = \frac{1}{(1 + \sigma^2)^2} \begin{bmatrix} 4(\sigma_1^2 - \sigma_2^2 - \sigma_3^2) + (1 - \sigma^2)^2 & 8\sigma_2\sigma_1 + 4\sigma_3(1 - \sigma^2) & 8\sigma_3\sigma_1 - 4\sigma_2(1 - \sigma^2) \\ 8\sigma_2\sigma_1 - 4\sigma_3(1 - \sigma^2) & 4(-\sigma_1^2 + \sigma_2^2 - \sigma_3^2) + (1 - \sigma^2)^2 & 8\sigma_3\sigma_2 + 4\sigma_1(1 - \sigma^2) \\ 8\sigma_3\sigma_1 + 4\sigma_2(1 - \sigma^2) & 8\sigma_3\sigma_2 - 4\sigma_1(1 - \sigma^2) & 4(-\sigma_1^2 - \sigma_2^2 + \sigma_3^2) + (1 - \sigma^2)^2 \end{bmatrix}.$$

## Используемые модели

Модель движения объекта наблюдения относительно инспектора в проекции на оси  $O_s x y z$  определяется уравнениями относительного движения, представленными в работе [7]

В свою очередь, вращательное движение объекта наблюдения в осях  $OXYZ$  определяется уравнениями, представленными в работе [8]. Единственным отличием при этом будет использование MRP в качестве параметров, описывающих относительную ориентацию.

Рассмотрим далее модель измерений оптического сенсора.

Модель камеры, используемая в данной работе, является классической для задач обработки измерений с оптических сенсоров [9], а преобразование пиксельных координат в компоненты, составляющие вектор визирования, впервые встречается в работах [2,10].

Выберем некоторую точку  $i$  на поверхности объекта наблюдения, радиус-вектор которой имеет вид  $r_{ССК}^i = [X_i, Y_i, Z_i]^T$  в ССК. В ССК эта точка имеет координаты  $r_{ССК}^i = [x_i^c, y_i^c, z_i^c]^T$ .

Геометрическая интерпретация представлена на рис. 3.

Проекцию особой точки на проецирующую плоскость сенсора без учета искажений можно записать как

$$u_i = -f(x_i^c/z_i^c);$$

$$v_i = -f(y_i^c/z_i^c)$$

или в развернутом виде:

$$u_i = -f \frac{A_{11}(X_i - X_c) + A_{12}(Y_i - Y_c) + A_{13}(Z_i - Z_c)}{A_{31}(X_i - X_c) + A_{32}(Y_i - Y_c) + A_{33}(Z_i - Z_c)},$$

$$v_i = -f \frac{A_{21}(X_i - X_c) + A_{22}(Y_i - Y_c) + A_{23}(Z_i - Z_c)}{A_{31}(X_i - X_c) + A_{32}(Y_i - Y_c) + A_{33}(Z_i - Z_c)},$$

где  $N$  — число измеряемых в один момент точек  $(x_i, y_i)$ ;

$(u_i, v_i)$  — координаты особой точки в ССК, пкс;

$(X_i, Y_i, Z_i)$  — координаты особой точки в ССК, м;

$(X_c, Y_c, Z_c)$  — неизвестные координаты оптического сенсора в ССК, м;

$f$  — фокусное расстояние, м;

$A_{ij}$  — компоненты матрицы перехода из ССК в ССК.

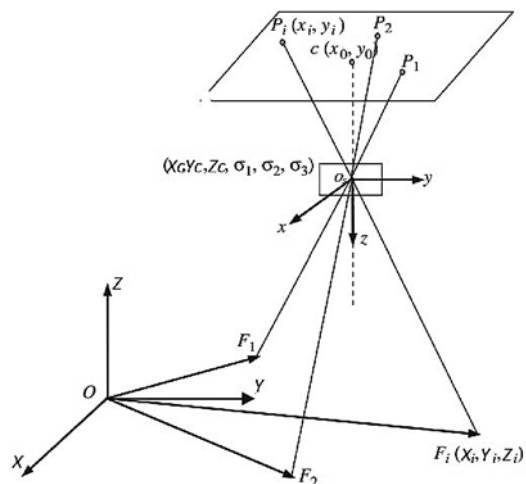


Рис. 3. Соотношения между системами координат и особые точки

Измерение в момент времени  $t$  определяется как

$$m_i \equiv \frac{1}{\sqrt{f^2 + x_i^2 + y_i^2}} \begin{bmatrix} x_i \\ y_i \\ -f \end{bmatrix}.$$

Вместе с тем, измерение в момент времени  $t$  можно записать как набор единичных векторов в форме

$$m_i = A r_i, \quad i = 1, 2, \dots, N,$$

$r_i$  — радиус-вектор, который имеет вид

$$r_i = \frac{1}{\sqrt{(X_i - X_c)^2 + (Y_i - Y_c)^2 + (Z_i - Z_c)^2}} \times \begin{bmatrix} (X_i - X_c) \\ (Y_i - Y_c) \\ (Z_i - Z_c) \end{bmatrix}.$$

Благодаря внутренним особенностям камеры реальные измерения оказываются зашумлены.

Модель измерений с учетом данного шума можно записать как

$$\tilde{b}_i = m_i(x) + v_i,$$

где  $\tilde{b}_i$  — измеренный вектор линии визирования  $i$  особой точки;

$x = [p, \sigma]^T$  — вектор состояния, содержащий относительное расстояние и относительную ориентацию;

$m_i(x) = A(\sigma_i)r_i$  — вектор линии визирования, построенный без учета шума.

Шум навигационный камеры моделируется посредством нормального распределения с ну-

левым математическим ожиданием и ковариационной матрицей ошибок измерений  $R_i$ :

$$E[v_i] = 0,$$

$$E[v_i v_j^T] = R_i \delta_{ij}.$$

### Метод обработки оптических изображений

Для определения относительной ориентации и положения объекта используем генетический алгоритм дифференциальной эволюции [1–13].

Принцип работы DE заключается в следующем.

#### 1. Создание популяции

В качестве начальной популяции выбирается случайный набор из  $N$  векторов из пространства  $R^n$ .

#### 2. Оценка популяции

Проводится оценка популяции с помощью функции рассогласования. Следует отметить, что конкретный вид данной функции будет выбран в рамках текущего исследования.

#### 3. Сортировка популяции

Популяция упорядочивается по условию минимума полученной оценки функции рассогласования (проводится сортировка массива с векторами популяции по минимуму функции рассогласования).

#### 4. Создание новых векторов популяции.

На очередном шаге алгоритма выполняется скрещивание каждой особи  $X$  из исходной популяции со случайно выбранной особью  $C$ , отличной от  $X$ . Координаты векторов  $X$  и  $C$  рассматриваются как генетические признаки. Перед скрещиванием применяется специальный оператор мутации — в скрещивании участвуют не исходные, а искаженные генетические признаки особи  $C$ :

$$C'_j = C_j + F(A_j - B_j),$$

где  $C'_j$  — искаженный признак вектора  $C$ ;  $C_j$  — исходный признак вектора  $C$ ;  $A_j, B_j$  — признак, выбранный случайно из сгенерированных в п. 1 векторов популяции;  $F$  — возмущение, вносимое в значение  $C'_j$ .

На данном этапе особое внимание следует уделить параметру  $F$ . Значение параметра является сугубо индивидуальным для каждой задачи и выступает в качестве предмета исследования текущей работы. Значение выбирается из диапазона, предложенного в работе [14].

#### 5. Создание новых векторов популяции (скрещивание).

Скрещивание выполняется следующим образом. Задается вероятность  $P$ , с которой потомок  $T$  наследует очередной (искаженный мутацией) генетический признак от родителя  $C$ . Соответствующий признак от родителя  $X$  наследуется с вероятностью  $(1 - P)$ . Фактически  $n$  раз разыгрывается бинарная случайная величина с математическим ожиданием  $P$ , и наследуется искаженный генетический признак от родителя  $C$  или генетический признак от родителя  $X$ . В результате формируется вектор-потомок  $T$ .

#### 6. Отбор векторов.

После формирования вектора-потомка  $T$  проводится сравнение функции рассогласования для него и для его "прямого" родителя  $X$ . В новую популяцию переносится тот из векторов  $X$  и  $T$ , на котором целевая функция достигает меньшего значения. Здесь необходимо заметить, что описанное правило отбора гарантирует неизменность размера популяции в процессе работы алгоритма.

Схема работы алгоритма представлена на рис. 4.

Для валидации полученных данных будет использоваться алгоритм, основанный на нелинейном методе наименьших квадратов (NLS).

Нелинейный метод наименьших квадратов — математический метод, применяемый для решения различных задач, основанный на минимизации суммы квадратов отклонений некоторых функций от искомым переменных. NLS является одним из базовых методов регрессионного анализа для оценки неизвестных параметров регрессионных моделей по выборочным данным.

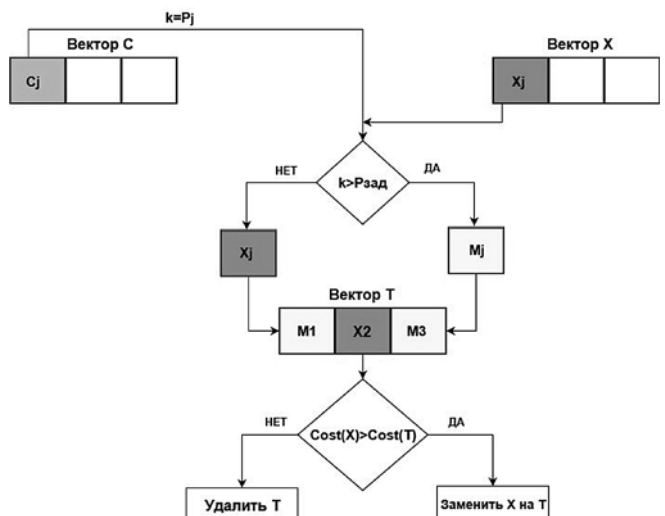


Рис. 4. Схема работы DE

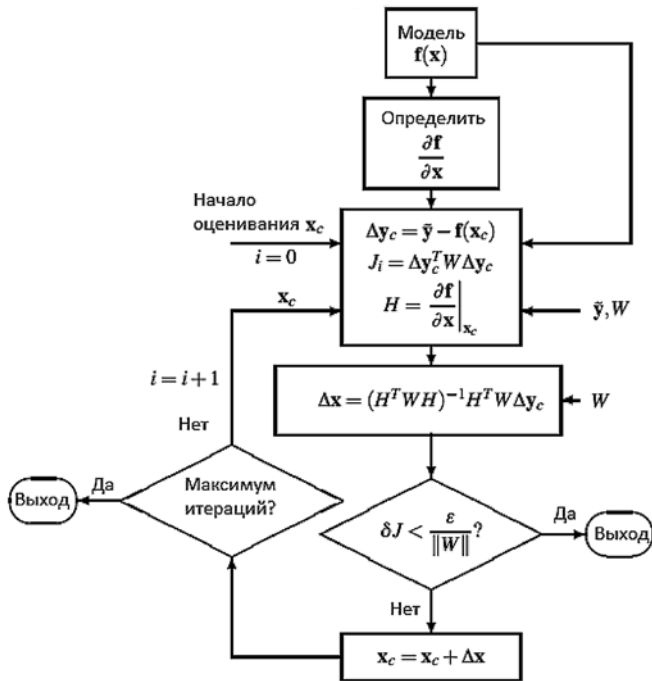


Рис. 5. Структурная схема алгоритма

Структурная схема алгоритма приведена на рис. 5.

Как видно из рис. 5, для реализации подхода необходимо знать также производные модели измерений по переменным вектора состояния.

Также необходимо знание о матрице весовых коэффициентов. Условимся, что матрица весовых коэффициентов имеет вид диагональной матрицы  $W$  со значениями  $w_{ij} = 10^{-6}$  на главной диагонали.

### Модельная задача

В качестве примера взят численный эксперимент по определению относительного положения и ориентации орбитальной ступени (ОС) с наноспутника при отделении его от ОС, выполняемом на типовой для наноспутников высоте полета и при небольших угловых скоростях. Разделение происходит по продольной оси объектов. Время моделирования составляет 20 с, временной интервал между двумя измерениями 0,1 с. Схема разделения представлена на рис. 6.

Параметры движения ОС относительно наноспутника будут определяться посредством шести ориентиров (без учета случая варьирования числа ориентиров). Определение относительных параметров движения в рамках задачи стартует только после полной видимости всех ориентиров на кадре, сделанном оптиче-

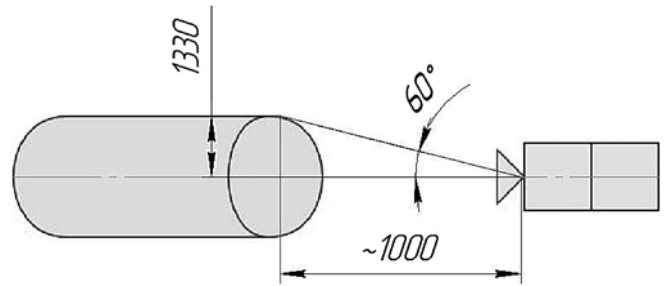


Рис. 6. Схема разделения

ским сенсором (на расстоянии примерно 1 м по программе отделения). Ниже приведены высота орбиты разделения, угловые скорости, сообщаемые при разделении, массоинерционные и габаритные характеристики ОС.

#### Параметры орбиты

Высота, км. . . . . 400

#### Параметры разделения

Угловые скорости, %/с

$\omega_x$  . . . . . 0

$\omega_y$  . . . . . 4

$\omega_z$  . . . . . 4

Линейные скорости, м/с

$V_x$  . . . . . 1.0

$V_y$  . . . . . 0

$V_z$  . . . . . 0

#### Массоинерционные и габаритные характеристики

$I_x$  . . . . . 2396,9

$I_y, I_z$  . . . . . 11470

Длина, м . . . . . 6,745

Масса, кг . . . . . 2710

В табл. 1 приведены данные о положении рассматриваемых на ОС ориентиров, на рис. 7 — габаритные размеры объекта наблюдения.

Таблица 1

Положение ориентиров в метрах

№	1	2	3	4	5	6
X, м	3,373	3,373	3,373	3,373	3,373	3,373
Y, м	0	1,33	0	0	-1,33	1,33
Z, м	0	0	1,33	-1,33	0	1,33

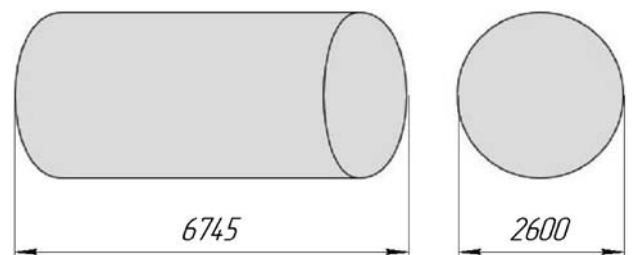


Рис. 7. Габаритные размеры объекта наблюдения

## Исследование гиперпараметров алгоритма дифференциальной эволюции

Для оценки вероятностных величин, используемых в алгоритме, используется метод статистических испытаний. Число испытаний определяется из соотношения [15]

$$N = \frac{1}{\varepsilon^2},$$

где  $N$  — число проведенных испытаний;  $\varepsilon$  — требуемая точность оценки вероятностных величин.

Так, для достижения 5 %-ной точности требуется провести не менее 400 испытаний.

Математическое ожидание вероятностной величины оценивается по соотношению

$$M_x = \frac{\sum_{i=1}^N X_i}{N},$$

где  $N$  — число проведенных испытаний;  $X_i$  — значение вероятностной величины;  $M_x$  — математическое ожидание вероятностной величины.

Среднеквадратическое отклонение находится по формуле

$$\sigma = \sqrt{\frac{\sum_{i=1}^N (X_i - M_x)^2}{n - 1}}.$$

Рассмотрим, при какой вероятности скрещивания удается добиться наилучшего решения.

Будем оценивать ошибку, полученную алгоритмом на каждом шаге, и рассмотрим, как меняется ошибка с ростом вероятности скрещивания.

Результаты 400 моделирований на интервале 20 с с шагом 0,1 с и шагом 0,1 по пороговому значению вероятности скрещивания  $P$  представлены в табл. 2 (см. вторую сторону обложки).

Следует заметить, что синий цвет соответствует меньшим значениям ошибки, а красный — большим.

Четко прослеживается следующая тенденция. При увеличении вероятности скрещивания уменьшается ошибка при решении задачи на заданном интервале, соответственно лучшими будут достаточно высокие значения пороговой вероятности.

Далее определимся, при какой степени мутации на этапе генерации новых векторов популяции удается добиться лучшего решения.

Результаты 400 моделирований на интервале 20 с с шагом 0,1 с и шагом 0,1 по степени

мутации представлены в табл. 3 (см. вторую сторону обложки).

Минимум ошибки наблюдается в табл. 3 в интервале 0,4...0,9, в качестве минимального значения из этого интервала можно выделить 0,8.

Оценим также число реперных точек на поверхности объекта и определим моментные характеристики для каждого набора точек.

Результаты 400 моделирований на интервале 20 с с шагом 0,1 с представлены в табл. 4, 5.

Оценим также время выполнения одной итерации. Гистограмма со средним временем исполнения одной итерации по 400 моделированиям приведена на рис. 8. Выполнение проводилось на стенде с характеристиками: Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz.

Отчетливо виден рост времени выполнения одной итерации (с 1,6 мс до 136,4 мс) при росте числа реперных точек (с 10 до 1000 точек), а следовательно, одномоментных измерений.

Таблица 4

Максимальное значение средней ошибки

Число точек	Среднее значение ошибки $\bar{\Delta}$					
	$\bar{\Delta}_\psi$	$\bar{\Delta}_\theta$	$\bar{\Delta}_\phi$	$\bar{\Delta}_x$	$\bar{\Delta}_y$	$\bar{\Delta}_z$
10	0,349	0,221	0,550	0,576	0,574	0,438
100	0,142	0,232	0,203	0,234	0,373	0,226
1000	0,023	0,034	0,039	0,173	0,160	0,212

Таблица 5

Среднеквадратическое отклонение

Число точек	Среднеквадратическое отклонение $\sigma$					
	$\sigma_\psi$	$\sigma_\theta$	$\sigma_\phi$	$\sigma_x$	$\sigma_y$	$\sigma_z$
10	0,02	0,022	0,023	0,077	0,072	0,081
100	0,03	0,024	0,012	0,038	0,035	0,028
1000	0,008	0,014	0,006	0,022	0,015	0,018

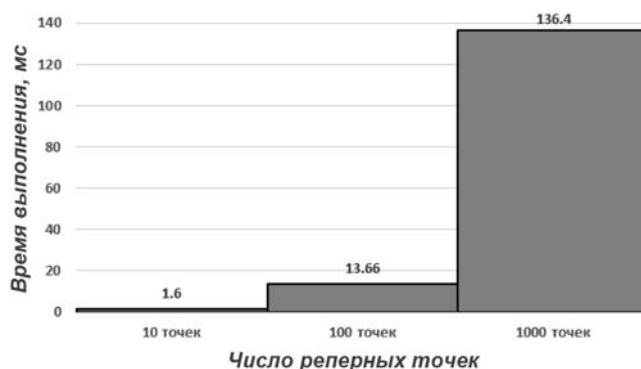


Рис. 8. Время выполнения одной итерации алгоритма для различного набора реперных точек

## Сравнение результатов работы алгоритма дифференциальной эволюции с нелинейным методом наименьших квадратов

Получим теперь оценки точности для алгоритма дифференциальной эволюции в сравнении с нелинейным методом наименьших квадратов, описанным ранее.

В качестве исходных данных будем также использовать модельную задачу разделения наноспутника и ОС в варианте с шестью реперными точками, представленными в табл. 2 (см. вторую сторону обложки). Определимся с начальными условиями алгоритмов.

Начальные условия DE:

$$bounds = [(0,50);(0,20);(0,20);(-0.1,0.1);(-0.1,0.1);(-0.1,0.1)];$$

фокусное расстояние  $f = 0,05$  м, пороговое значение вероятности скрещивания  $P = 0,97$ , размер популяции  $size = 20$  особей, максимальное число итераций  $iter = 200$ , степень мутации  $F = 0,8$ .

Начальные условия NLS:

$$bounds = [3.5,2.6,2.6,1e-6,1e-6,1e-6].$$

Результаты 400 моделирований на интервале 20 с с шагом 0,1 с представлены на рис. 9, 10.

Анализируя графики, можно сделать вывод, что ошибка для DE при решении модельной задачи не превышает  $0,05^\circ$ , тогда как у NLS она составляет не более  $0,5^\circ$ . Аналогичная ситуация проявляется и для ошибки в определении относительного положения. Ошибка для DE при решении модельной за-

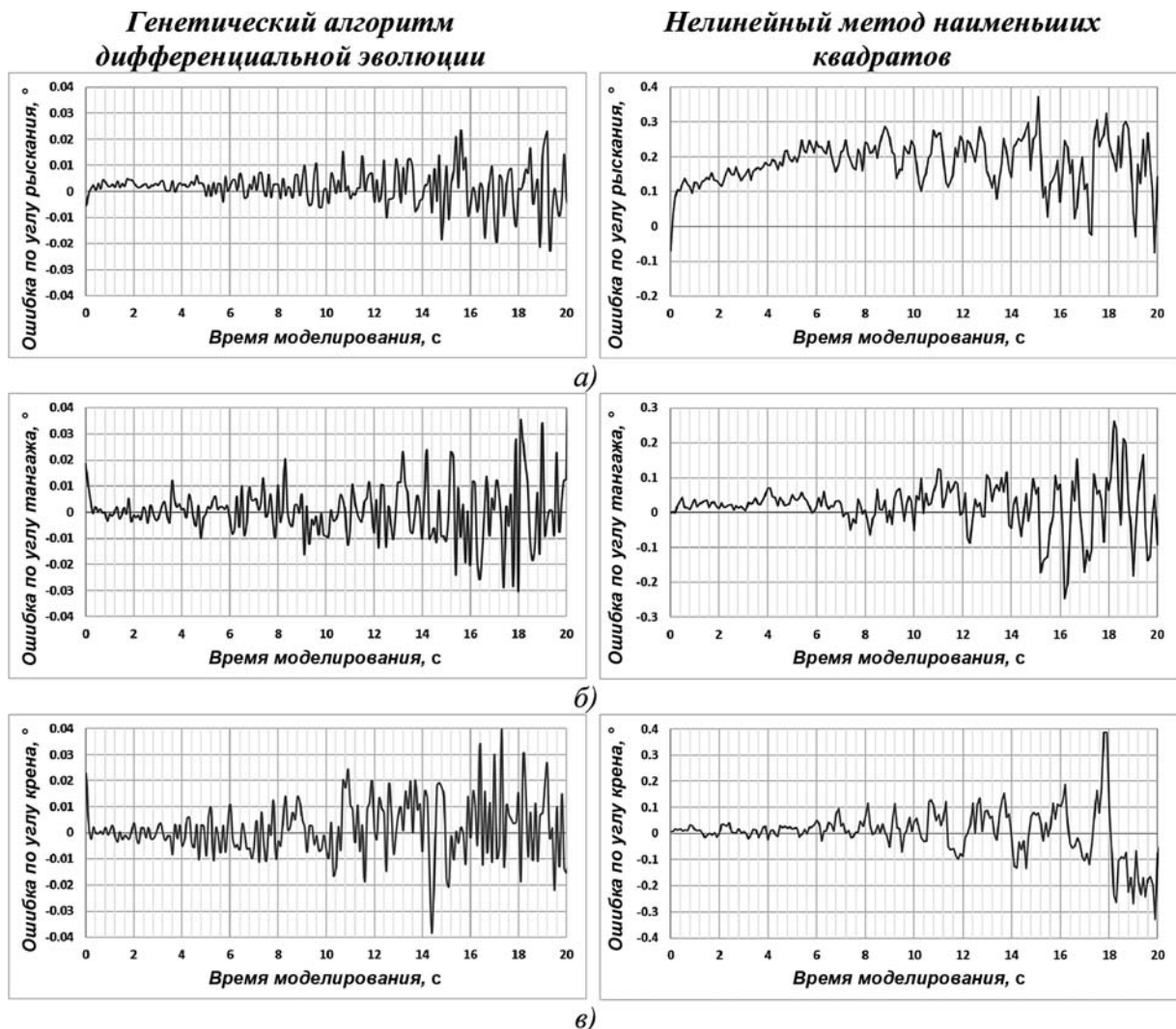


Рис. 9. Ошибки определения относительной ориентации при решении модельной задачи для двух алгоритмов: а — ошибка по углу рыскания; б — ошибка по углу тангажа; в — ошибка по углу крена

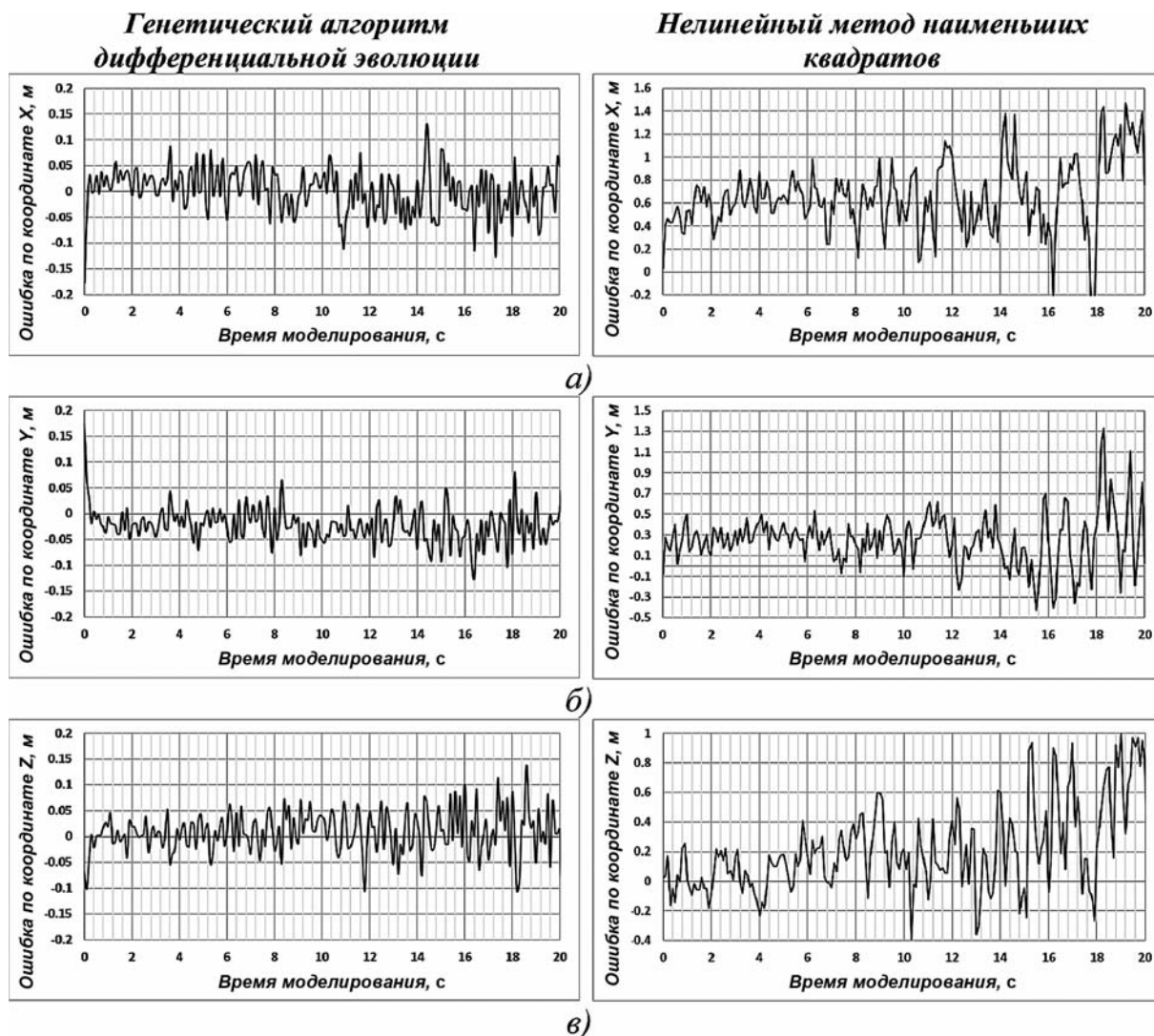


Рис. 10. Ошибки определения относительного положения при решении модельной задачи для двух алгоритмов: а — ошибка по координате  $X_c$ ; б — ошибка по координате  $Y_c$ ; в — ошибка по координате  $Z_c$

дачи не превышает 0,2 м, а для NLS ошибка составляет 1...2 м.

### Заключение

В рамках работы было проведено статистическое исследование генетического алгоритма дифференциальной эволюции, включающее в себя исследование влияния параметров настройки алгоритма (гиперпараметров) на точность получаемого результата. Было выявлено, что с ростом числа одномоментных измерений точность алгоритма увеличивается: средняя ошибка по углам относительной ориентации уменьшается с десятых долей градуса до сотых долей градуса, а по компонентам относительного расстояния уменьшается с 0,6 м до 0,1 м соответственно. При этом с ростом точности

значительно растет и время выполнения одной итерации: с 1,6 мс до 136,4 мс соответственно.

Также было проведено сравнение решения модельной задачи с помощью двух алгоритмов (DE и NLS) для шести реперных точек на поверхности объекта наблюдения.

Выявлено, что при схожих условиях DE показывает точность решения, большую на один порядок. Например, ошибка DE при решении модельной задачи не превышает 0,05°, тогда как у NLS она составляет не более 0,5°. Похожая ситуация проявляется и для ошибки в определении относительного положения: ошибка DE при решении модельной задачи не превышает 0,2 м, а для NLS ошибка составляет 1...2 м.

Исследование показывает, что генетический алгоритм дифференциальной эволюции можно применять в различных задачах обработки информации с оптических датчиков.



## Список литературы

1. Lee D., Pernicka H. Vision-Based Relative State Estimation Using the Unscented Kalman Filter // *Int'l J. of Aeronautical & Space Sci.* 2011. Vol. 12. P. 24–36.
2. Junkins J. L., Hughes D. C., Wazni K., Parriyapong V. Vision based navigation for rendezvous, docking and proximity operations // *22nd Annual AAS Guidance and Control Conference*. Breckenridge, 1999. AAS 99-021.
3. Tang X., Zhong D. Square-root sigma-point Kalman filtering for spacecraft relative navigation // *Acta Astronautica*, 2010. Vol. 66. P. 704–713.
4. Борисов А. В., Караваев Ю. Л. Экспериментальное исследование движения тела с осесимметричным основанием, скользящего по шероховатой поверхности // *Нелинейная динамика*. 2015. Т. 11. № 3. С. 547–577.
5. Sabatini M. A testbed for visual based navigation and control during space rendezvous operations // *Acta Astronautica*. 2015. N.117. P. 184–196.
6. Schaub H., Junkins J. L. *Analytical Mechanics of Space Systems*. AIAA education series, 2009. P. 795.
7. Апазов Р. Ф., Сытин О. Г. Методы проектирования траекторий носителей и спутников Земли. М.: Наука, 1987. 440 с.
8. Попов В. И. Системы ориентации и стабилизации космических аппаратов. М: Машиностроение, 1986. 184 с.
9. Junkins J. L., Crassidis J. L. *Optimal Estimation of Dynamic Systems*. Taylor and Francis Group LLC, 2012. P. 733.
10. Gunnam K. K., Hughes D. C., Junkins J. L., Kehtarnavaz N. A vision based DSP embedded navigation sensor // *IEEE Sensor Journal*. 2002. Vol. 5. P. 428–442.
11. Price K. *Differential Evolution — A Practical Approach to Global Optimization*. Berlin: Springer, 2005. P. 539.
12. Сабанин В. Р., Репин А. И. Модифицированный генетический алгоритм для задач оптимизации в управлении // *Exponenta pro: математика в приложениях*, 2004. № 3–4. С. 78–85.
13. Кирсанов М. Н. Генетический алгоритм оптимизации стержневых систем // *Строительная механика и расчет сооружений*. 2010. № 2 (229). С. 60–63.
14. Storn R., Price K. *Differential Evolution — A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces* // *Journal of Global Optimization*. 1997. V. 11. P. 341–359.
15. Белоконов И. В. Статистический анализ динамических систем (анализ движения летательных аппаратов в условиях статистической неопределенности). Самара: Самарский госуд. аэрокосм. ун-т, 2001.

S. P. Simakov, Postgraduate Student, e-mail: simba1393@gmail.com,  
Samara University

## Statistical Study of the Differential Evolution Algorithm for Processing Optical Measurements

*In this article were discussed the using of the genetic algorithm of differential evolution to determine the orientation and position of an object relative to an optical sensor by means a series of frames. In this case, commercial optical cameras are used as measuring means. A statistical study of the genetic algorithm of differential evolution was conducted. The study was conducted on the model problem of separation of two objects. Validation of estimates is carried out using the nonlinear least squares method. The paper presents the dependencies of errors in determining the relative orientation and relative position obtained using two algorithms. The study shows that the genetic algorithm of differential evolution can be used in various problems of processing information from optical sensors.*

**Keywords:** nanosatellite, least squares algorithm, optical sensor, relative motion parameters, differential evolution algorithm, inspector, observation object

DOI: 10.17587/it.26.410-418

### References

1. Lee D., Pernicka H. Vision-Based Relative State Estimation Using the Unscented Kalman Filter, *Int'l J. of Aeronautical & Space Sci.*, 2011, vol. 12, pp. 24–36.
2. Junkins J. L., Hughes D. C., Wazni K., Parriyapong V. Vision based navigation for rendezvous, docking and proximity operations, *22nd Annual AAS Guidance and Control Conference*, Breckenridge, 1999, AAS 99-021.
3. Tang X., Zhong D. Square-root sigma-point Kalman filtering for spacecraft relative navigation, *Acta Astronautica*, 2010, vol. 66, pp. 704–713.
4. Borisov A. V., Karavaev Yu. L. An experimental study of the motion of a body with an axisymmetric base sliding on a rough surface, *Nonlinear dynamics*, 2015, vol. 11, no. 3, pp. 547–577 (in Russian).
5. Sabatini M. A testbed for visual based navigation and control during space rendezvous operations, *Acta Astronautica*, 2015, no. 117, pp. 184–196.
6. Schaub H., Junkins J. L. *Analytical Mechanics of Space Systems*, *AIAA Education Series*, 2009, p. 795.
7. Apazov R. F., Sytin O. G. Methods of designing the trajectories of a satellites, Moscow, Nauka, 1987, 440 p. (in Russian).
8. Popov V. I. Spacecraft orientation and stabilization systems, Moscow, Mashinostroenie, 1986, 184 p. (in Russian).
9. Junkins J. L., Crassidis J. L. *Optimal Estimation of Dynamic Systems*, Taylor and Francis Group LLC, 2012, pp. 733.
10. Gunnam K. K., Hughes D. C., Junkins J. L., Kehtarnavaz N. A vision based DSP embedded navigation sensor, *IEEE Sensor Journal*, 2002, vol. 5, pp. 428–442.
11. Price K. *Differential Evolution — A Practical Approach to Global Optimization*, Berlin, Springer, 2005, 539 p.
12. Sabanin V. R., Repin A. I. Modified genetic algorithm for optimization problems in guidance, *Exponenta Pro: Matematika v Prilozheniyah*, 2004, no. 3–4, pp. 78–85 (in Russian).
13. Kirsanov M. N. Genetic algorithm for optimizing pivot systems, *Stroitel'naya Mekhanika i Raschet Sooruzhenij*, 2010, no. 2 (229), pp. 60–63 (in Russian).
14. Storn R., Price K. *Differential Evolution — A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces*, *Journal of Global Optimization*, 1997, vol. 11, pp. 341–359.
15. Belokonov I. V. Statistical analysis of dynamical systems (analysis of aircraft movement under conditions of statistical uncertainty, Samara, Samara state aerosp. univ., 2001 (in Russian).

**«СТАТИСТИЧЕСКОЕ ИССЛЕДОВАНИЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДИФФЕРЕНЦИАЛЬНОЙ ЭВОЛЮЦИИ ДЛЯ ОБРАБОТКИ ОПТИЧЕСКИХ ИЗОБРАЖЕНИЙ»**

Таблица 2

Ошибка между истинными и найденными значениями для различных вероятностей скрещивания и заданной программе отделения

Время, с	Порог вероятности скрещивания $P$									
	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
0	239,80	154,94	251,44	2,76	180,92	120,84	181,52	394,64	194,07	174,62
1	161,20	315,20	470,49	7,50	390,59	335,57	147,31	492,09	41,53	112,92
2	209,31	197,79	390,55	11,61	340,60	223,94	135,58	341,64	1,31	37,96
3	78,74	230,99	299,59	18,46	32,19	168,26	4,54	279,57	1,71	16,59
4	36,60	94,46	212,00	17,88	116,27	14,01	1,55	207,32	1,39	0,00
5	66,88	58,89	106,80	65,61	55,97	20,07	0,93	47,84	1,40	0,04
6	102,91	84,78	97,48	42,76	57,27	6,44	17,09	7,17	1,38	0,02
7	32,60	6,98	39,12	137,37	4,79	21,28	29,45	14,67	1,08	0,03
8	18,63	180,67	160,29	47,49	49,08	210,89	90,73	13,74	1,54	0,15
9	132,91	29,64	271,29	7,47	56,13	50,55	5,79	24,29	23,54	0,46
10	303,12	32,53	81,58	32,36	40,67	577,11	4,75	17,52	1,89	0,02
11	344,87	70,77	193,05	30,58	51,58	195,79	10,68	87,03	3,97	0,04
12	69,92	16,01	45,22	79,27	140,57	110,18	60,70	112,38	3,29	0,12
13	60,78	220,87	0,34	390,01	40,58	8,02	70,85	432,45	43,62	0,01
14	233,47	36,86	55,66	469,52	28,53	69,21	93,25	121,50	17,24	0,24
15	12,13	136,96	114,48	165,05	34,08	2,18	54,04	124,21	34,66	2,13
16	118,17	23,06	47,85	125,85	28,27	78,52	65,92	181,07	11,15	0,48
17	25,32	159,96	91,71	155,98	18,59	123,93	59,21	210,35	16,24	6,66
18	40,57	418,16	13,34	254,55	18,41	83,28	32,93	118,02	18,29	0,29
19	28,87	571,46	51,40	248,24	58,52	241,24	32,65	25,97	11,96	10,09

Таблица 3

Ошибка между истинными и найденными значениями для различных степеней мутации и заданной программе отделения

Время, с	Степень мутации										
	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1	1,1
0	127,6	316,8	104,1	190,912	118,815	231,876	203,508	223,508	279,620	353,6	266,5
1	34,1	729,7	72,9	143,485	362,471	454,825	359,301	159,301	105,575	746,6	613,0
2	10,3	465,2	23,5	162,233	383,981	370,900	387,237	187,237	137,470	534,6	430,6
3	96,7	255,8	33,4	0,001	61,592	18,156	256,475	56,475	97,376	391,8	294,0
4	186,5	184,6	6,1	11,768	0,000	0,000	0,001	0,001	0,034	250,1	6,8
5	102,1	253,0	497,1	0,005	0,004	0,001	0,006	0,003	0,008	177,7	3,2
6	18,4	141,9	249,3	1,119	0,002	0,003	0,001	0,006	0,015	122,4	3,8
7	188,8	193,9	52,0	0,340	0,004	0,003	0,000	0,003	0,071	0,0	2,6
8	498,1	335,9	44,2	2,961	0,021	0,001	0,001	0,003	0,008	1,7	27,7
9	541,7	477,5	60,8	0,001	0,004	0,004	0,022	0,004	0,077	333,5	0,2
10	839,9	200,3	46,0	1,359	0,002	0,004	0,002	0,005	0,131	9,0	1,4
11	1128,1	169,1	116,1	2,710	0,064	0,001	0,005	0,050	0,036	3,1	43,2
12	1219,1	8,1	13,4	0,013	0,128	0,032	0,017	0,052	0,153	2,9	0,3
13	1178,0	27,0	4,9	0,054	0,035	0,115	0,029	0,011	0,060	11,3	5,0
14	1200,0	15,5	1,4	0,660	0,011	0,025	0,052	0,012	0,097	1,4	54,4
15	1231,0	17,9	7,4	5,241	0,207	0,027	0,007	0,009	0,902	17,6	136,2
16	1280,0	62,4	10,2	3,174	0,000	0,258	0,110	0,051	0,276	22,4	28,2
17	1345,0	235,8	21,3	1,289	0,006	0,029	0,018	0,030	0,906	41,4	49,2
18	1432,0	320,1	87,5	2,902	0,138	0,091	0,219	0,527	1,094	1,5	180,8
19	1480,0	285,5	118,6	1,123	0,177	0,098	0,295	0,082	1,158	3,0	13,2