

DOI: <https://doi.org/10.30898/1684-1719.2025.8.8>

УДК: 004.942; 004.422.8; 621.396.96

## АДАПТАЦИЯ МОДЕЛИ ИНДИКАТОРА КРУГОВОГО ОБЗОРА РАДИОЛОКАЦИОННОЙ СТАНЦИИ

С.В. Нестеров<sup>1</sup>, С.А. Рутковский<sup>2</sup>

<sup>1</sup> Филиал «Взлёт» Московского Авиационного Института,  
416501, Астраханская область, г. Ахтубинск, ул. Добролюбова, 5

<sup>2</sup> Научно-исследовательский институт приборостроения имени В.В. Тихомирова,  
140180, Московская область, город Жуковский, ул. Гагарина, д. 3

Статья поступила в редакцию 18 мая 2025 г.

**Аннотация.** В филиале «Взлет» МАИ в г. Ахтубинске в программной среде MATLAB ранее была разработана компьютерная модель индикатора кругового обзора радиолокационной станции. Данная модель используется при изучении студентами дисциплин таких как «основы теории радионавигационных систем», «основы теории радиолокационных систем», «основы теории радиоэлектронной борьбы», «основы теории систем радиоуправления». В результате приостановки деятельности компании The MathWorks (США) в России, были отозваны лицензии на использование пакета прикладных программ MATLAB. В статье описывается процесс адаптации модели под среду программирования Python. Представлены математические методы, используемые при адаптации, реализация модели в виде части программного кода, результаты экспериментов по моделированию радиолокационной обстановки, а также результаты сравнения двух моделей. Адаптированная модель индикатора кругового обзора обладает большим потенциалом дальнейшего развития по совершенствованию радиолокационной обстановки и приближению её к более реальным обстановкам.

**Ключевые слова:** радиолокационная станция кругового обзора, тренажёрный имитатор, радионавигация, радиолокация, Python, MATLAB.

**Автор для переписки:** Рутковский Сергей Алексеевич,  
[sergeirutcovsky@gmail.com](mailto:sergeirutcovsky@gmail.com)

## Введение

Моделирование в области радиолокации неоднократно рассматривалось в ряде работ [1], включая модели целей и помех [2]. Так и в филиале «Взлет» МАИ в г. Ахтубинске в программной среде MATLAB ранее была разработана компьютерная модель индикатора кругового обзора обзорной радиолокационной станции (РЛС) (рисунок 1) для использования в учебных целях [3].

Однако, компания The MathWorks (США) с 5 марта 2022 года приостановила деятельность в России и отозвала лицензии у российских студентов и университетов [4]. Чтобы имелась возможность дальнейшего использования разработанной модели, необходимо было адаптировать разработанную модель к сложившимся условиям, т.е. записать в новой среде, на ином языке программирования, который популярен среди программистов и имеется в широком доступе.

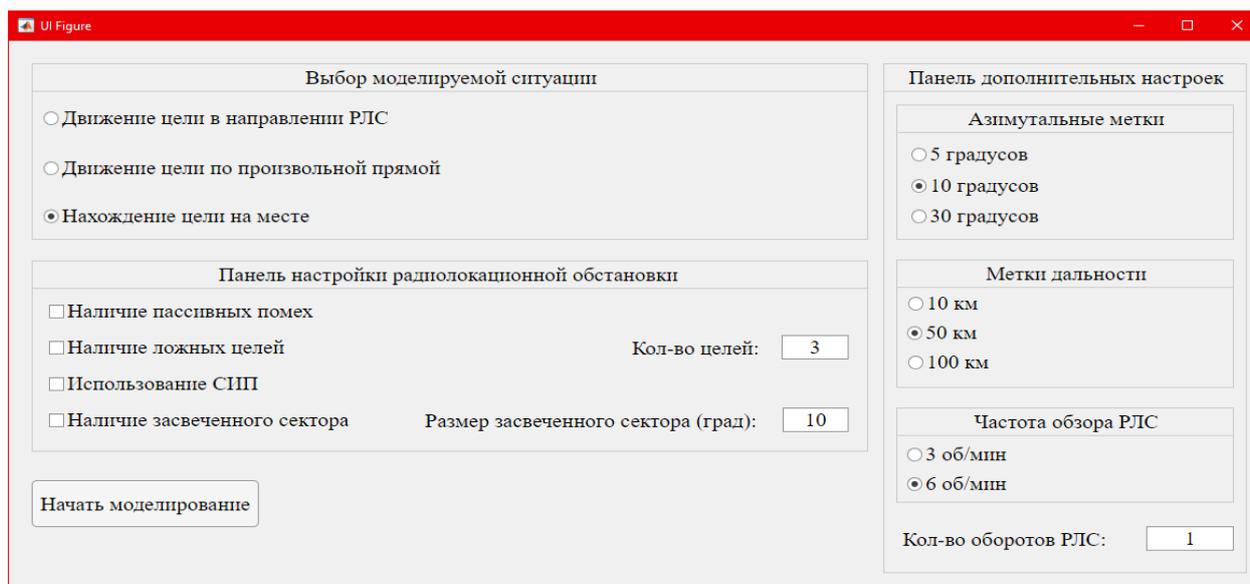


Рис. 1. Пользовательское окно модели индикатора в среде MATLAB.

## 1. Разработанная модель индикатора кругового обзора радиолокационной станции

В качестве новой среды разработки был выбран язык программирования Python. Данный язык программирования был выбран по ряду причин:

- 1) python имеет простой и понятный синтаксис, что делает его легким в изучении и использовании, особенно для начинающих;
- 2) данный язык программирования имеет большое кол-во библиотек, которые можно использовать для различных задач, в том числе библиотеку matplotlib, во многом копирующую инструменты пакета MATLAB;
- 3) python является языком с открытым исходным кодом, что означает, что его исходный код находится в свободном доступе и может быть изменен или распространен кем угодно.

Разработанная компьютерная модель индикатора радиолокационной станции, пользовательское окно которой представлено на рисунке 2, не только не утратила возможности предыдущей модели, но и приобрела ряд новых функций:

- 1) В панели «Выбор моделируемой ситуации» появилась возможность выбрать новую модель движения цели «Движение цели по ломаной прямой», в которой цель будет двигаться по ломаной прямой, состоящей из 3, 4 или 5 отрезков.
- 2) Появилась дополнительная панель «Скорость летательного аппарата», в которой можно задать скорость цели или оставить данный параметр случайным (скорость определится случайным образом в диапазоне от 300 до 3000 км/ч).
- 3) Появилось несколько кнопок отвечающие за следующие функции: приостановка моделирования для детального разбора обстановки на индикаторе (кнопка «Остановить моделирование»), продолжение моделирование после приостановки (кнопка «Продолжить моделирование») и завершение моделирования для досрочного завершения работы (кнопка «Закончить моделирование»).

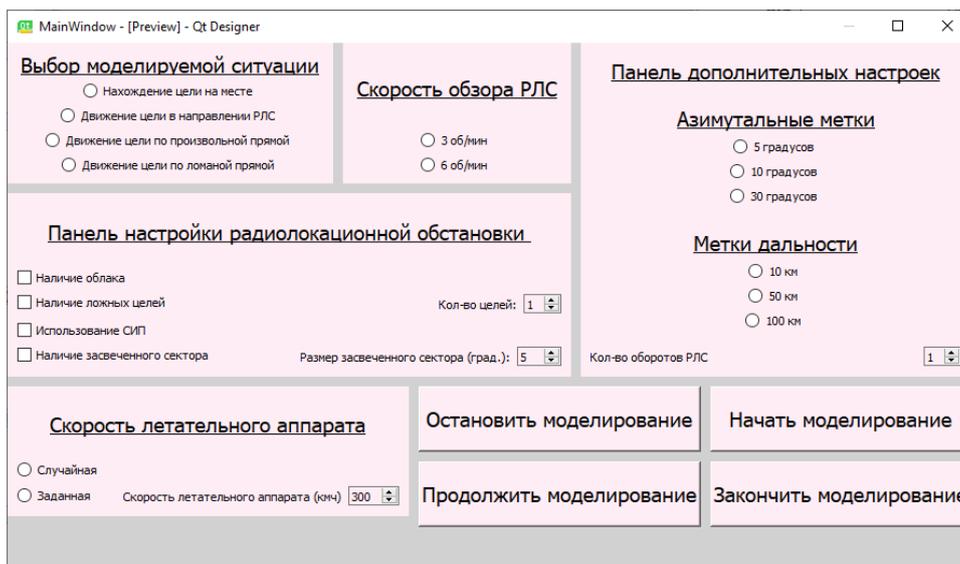


Рис. 2. Пользовательское окно компьютерной модели индикатора радиолокационной станции.

При нажатии кнопки «Запуск» производится моделирование работы индикатора РЛС в соответствии с теоретическими положениями радиолокации, радионавигации, радиоэлектронной борьбы и радиосистем управления (рис. 3). Важно отметить, что цвет моделируемого индикатора, как и реального индикатора, – желто-зелёный, в отличие от цвета индикатора предыдущей версии модели (рис. 4).

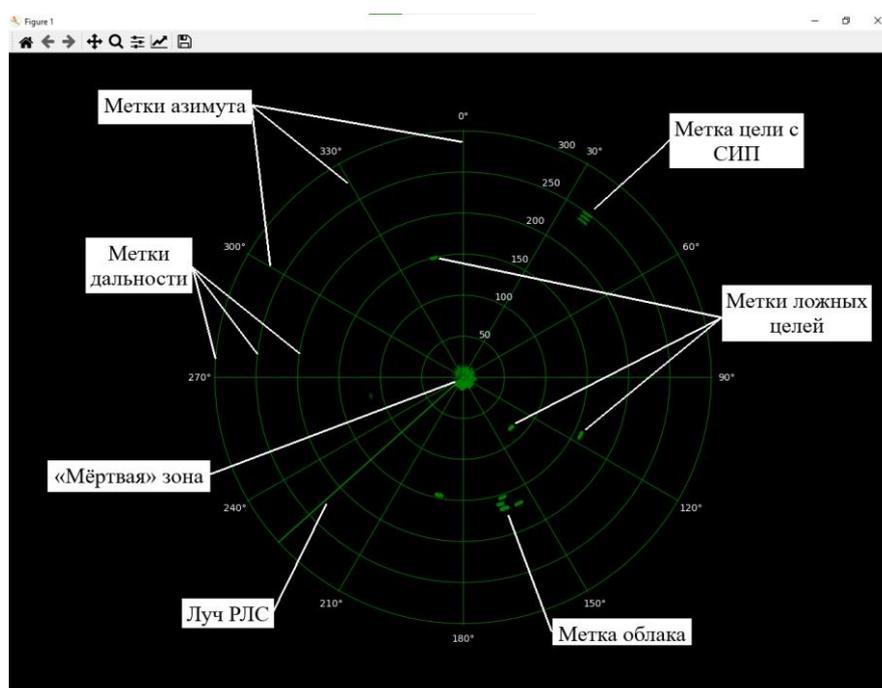


Рис. 3. Экран разработанной модели индикатора радиолокационной станции.

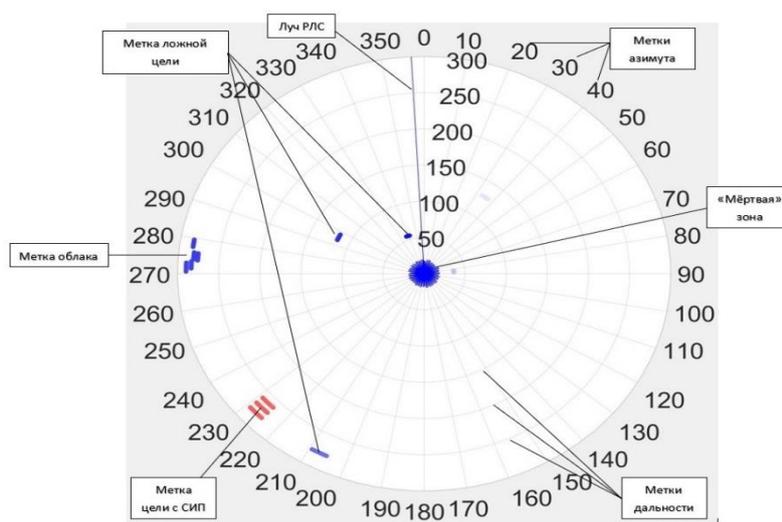


Рис. 4. Экран предыдущей модели индикатора кругового обзора радиолокационной станции.

## 2. Математические методы, используемые при адаптации

Наибольшие затруднения вызвали создание нового способа моделирования засвеченного сектора от активных помех и моделирование движения цели по ломаной прямой.

Засвеченный сектор индикатора РЛС возникает вследствие наличия активных помех в основном лепестке диаграммы направленности антенны (ДНА), что позволяет грубо определить направление на источник помех, однако дальность до него определить невозможно; также невозможно наблюдать отметки целей (рисунок 5) [5]. Такой сектор появляется в результате равномерного воздействия множества точечных электронов на участок люминофора индикатора. В предыдущей модели засвеченный сектор моделировался с помощью множества прямых исходящих из центра и отстоящих друг от друга на некоторый угол  $\alpha$  (рисунок 6, а). При этом формировалась равномерно закрашенная фигура (рисунок 6, б), что не соответствует действительности. В новой версии модели было решено использовать несколько множеств точек, сформированных в небольшие сектора шириной  $\beta$  (рисунок 7), для закрашивания необходимого участка окружности. Применение секторов обусловлено существенным замедлением работы модели, из-за большого числа сравнений текущего положения луча РЛС

и положения каждой точки засвеченного участка (число которых достигает нескольких тысяч при закрашивании сектора величиной  $30^\circ$ ).

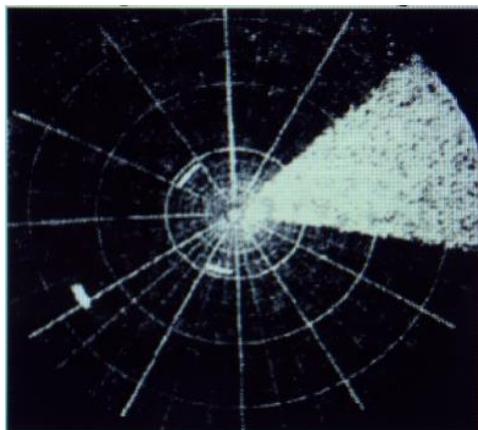


Рис. 5. Снимок реального индикатора радиолокационной станции с засвеченным сектором.

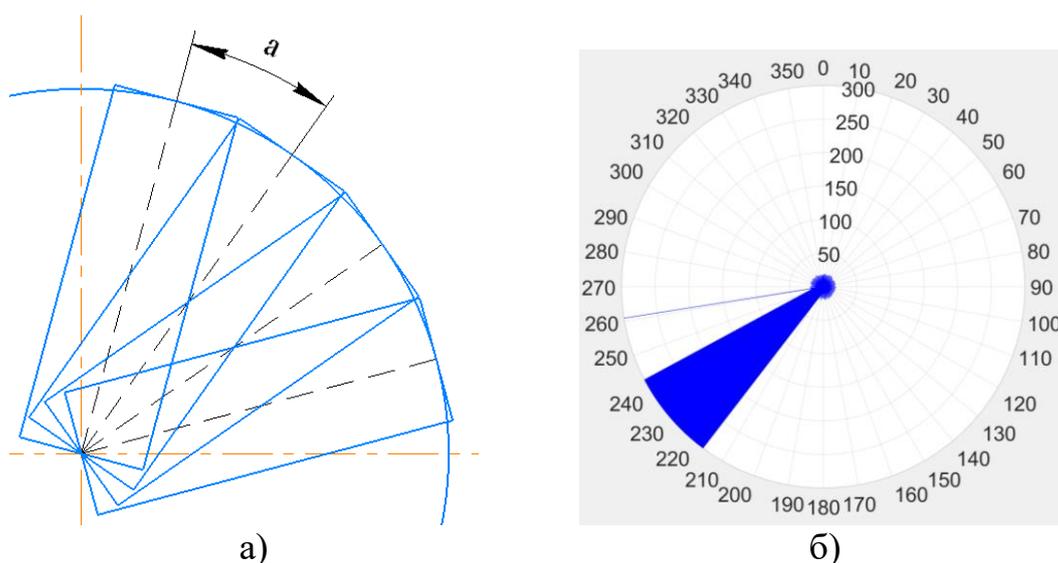


Рис. 6. Моделирование засвеченного сектора в предыдущей модели. а) к пониманию формирования; б) засвеченный сектор.

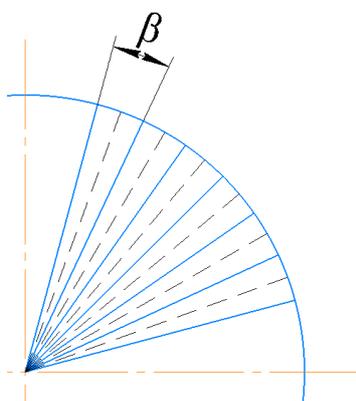


Рис. 7. К пониманию моделирования засветки.

Однако использование стандартной рандомизированной функции языка Python для формирования множества точек сектора не рационально. Данная функция имеет равномерное распределение вероятности по радиусу окружности, в то время как площадь сектора  $S$  имеет квадратичную зависимость от радиуса  $r$  этого сектора:

$$S(r) = \frac{\pi r^2}{360} \varphi, \quad (1)$$

где  $\varphi$  – угол сектора. В результате, плотность распределения точек в секторе не равномерна и по мере удаления от центра индикатора заметно уменьшается (рисунок 8).

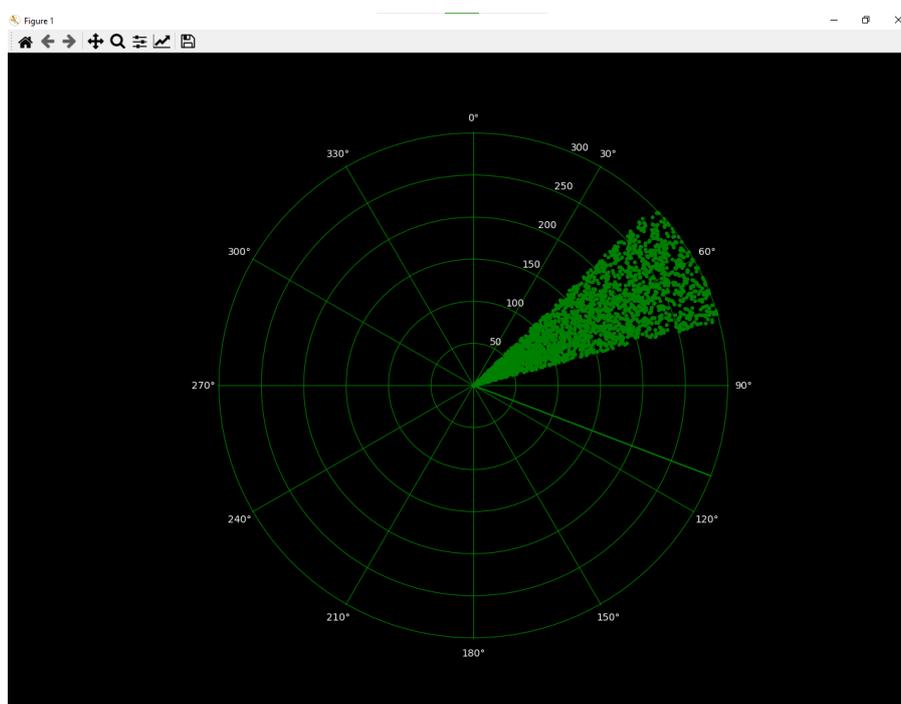


Рис. 8. Использование равномерного распределения вероятности для формирования засветки.

Для решения этой проблемы было решено использовать квадратичную часть нормального закона распределения плотности вероятности  $f(r)$ , определяемого по формуле:

$$f(r) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(r-m)^2}{2\sigma^2}\right], \quad (2)$$

где  $\sigma$  – среднее квадратическое отклонение,  $m$  – математическое ожидание. В данном случае с увеличением радиуса сектора количество генерируемых точек будет увеличиваться по квадратичному закону, как и площадь сектора, которую требуется закрасить. Экспериментальным путем было выяснено, что наиболее схожим с зависимостью площади сектора от радиуса  $S(r)$  на участке от 0 до 300 км является нормальное распределение вероятности со среднее квадратическим отклонением  $\sigma$ , равным 138 км, и математическим ожиданием  $m$ , равным 400 км (рисунок 9).

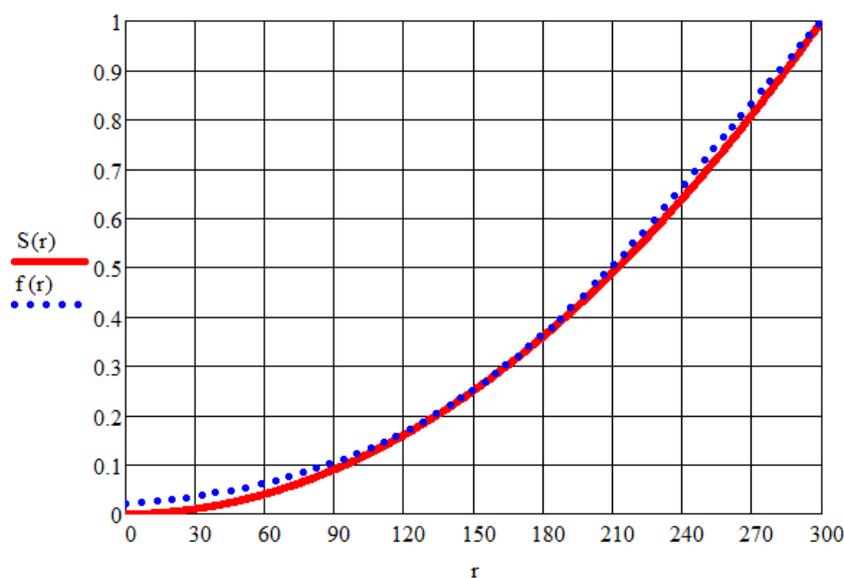


Рис. 9. Зависимости  $S(r)$  и  $f(r)$  на участке от 0 до 300 км в нормированном виде.

Полученный в результате такого моделирования сектор представлен на рисунке 10. Формирование засвеченного сектора в программном представлении изображено на рисунке 11.

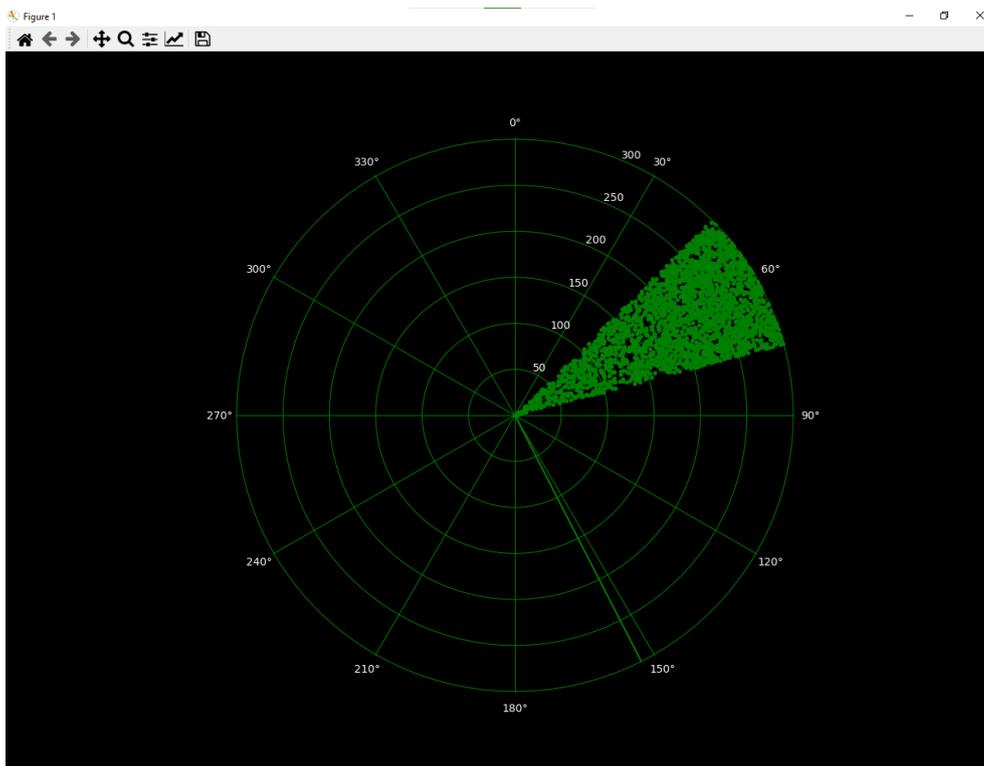


Рис. 10. Смоделированный засвеченный сектор.

```

ugol_zasv = 30 # Размер засвечиваемой области в градусах ;Z0
rad_ugol_zasveta=ugol_zasv/360*2*np.pi # Размер засвечиваемой области в радианах
razmer_sektora=1# размер сектора в градусах
n_t_sekt=70 # кол-во точек на сектор
r_zasveta=[[0 for j in range(n_t_sekt)]for i in range(ugol_zasv)]
for i in range(ugol_zasv):
    j=0
    while j < n_t_sekt:
        r_zasveta[i][j]=np.random.normal(loc= 400, scale= 138, size= 1)
        if r_zasveta[i][j]<300 and r_zasveta[i][j]>0:
            j=j+1
t_zasveta = [t_celi - rad_ugol_zasveta / 2 + random.random()*razmer_sektora
              * 2 * np.pi / 360 for j in range(n)]
marker_zasveta = [t_celi - rad_ugol_zasveta / 2 + (i+1) *razmer_sektora / 360
                  * 2 * np.pi for i in range(ugol_zasv)]

```

Рис. 11. Программное представление формирования засветки.

Для моделирования движения цели по ломаной линии, сначала необходимо определить уравнения прямых, из которых состоит данная ломаная. В качестве начальных данных используются полярные координаты  $R_i$  (полярный радиус) и  $\theta_i$  (полярный угол) точек пересечения этих прямых, а также конечных точек ломаной (рисунок 12).

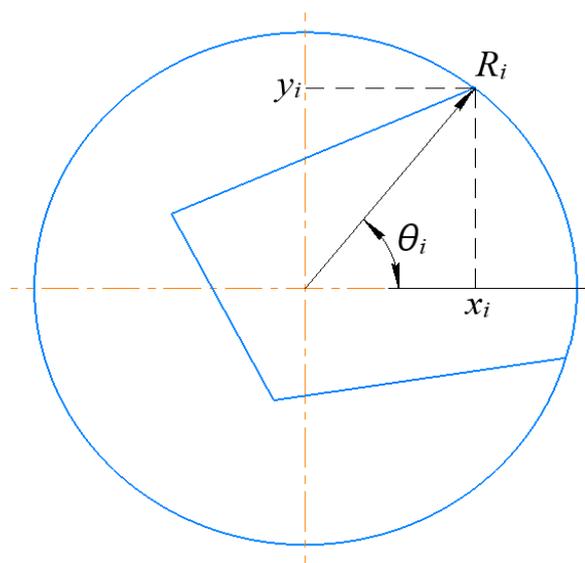


Рис. 12. К пониманию формирования траектории движения по ломаной линии.

Преобразуем полярные координаты точек в прямоугольные координаты, используя соответствующие формулы перехода:

$$\begin{aligned} x_i &= R_i \cos \theta_i, \\ y_i &= R_i \sin \theta_i, \end{aligned} \quad (3)$$

где  $x$ ,  $y$  – координаты по осям  $x$  и  $y$  соответственно.

Уравнение прямой в прямоугольной системе координат имеет вид:

$$y = kx + b, \quad (4)$$

где  $k$  – угловой коэффициент,  $b$  – ордината точки пересечения прямой с осью  $y$ .

Используя выражение (4) и набор прямоугольных координат точек  $x_i$ ,  $y_i$  получим выражения для определения коэффициентов  $k$  и  $b$  прямых, образующих ломаную:

$$\begin{cases} k_i = \frac{y_{i+1} - y_i}{x_{i+1} - x_i}, i = 1 \dots n - 1, \\ b_i = y_i - k_i x_i \end{cases} \quad (5)$$

где  $n$  – общее количество точек.

Для равномерного передвижения цели по заданной траектории необходимо, чтобы расстояние  $L$  проходимое целью за время одной итерации цикла модели было постоянным, т.е.:

$$(x_2 - x_1)^2 + (y_2 - y_1)^2 = L^2, \quad (6)$$

где  $x_1, y_1$  – предыдущие координаты цели,  $y_2$  – новые координаты цели.

Так как цель движется по прямой с известными коэффициентами  $k_i$  и  $b_i$ , то можно выразить  $y_2$  используя выражение (4):

$$(x_2 - x_1)^2 + (k_i x_2 + b_i - y_1)^2 = L^2. \quad (7)$$

В результате простых математических преобразований было получено квадратное уравнение относительно  $x_2$ :

$$(k_i^2 + 1)x_2^2 - 2(x_1 - k_i b_i - y_1)x_2 + x_1^2 + (b_i - y_1)^2 - L^2 = 0. \quad (8)$$

Решив данное уравнение, можно получить выражение, определяющее новую координату  $x_2$  цели, и как следствие координату  $y_2$ , в соответствии с условием равномерного движения:

$$x_2 = \frac{(x_1 - k_i b_i - y_1) \pm \sqrt{(x_1 - k_i b_i - y_1)^2 - (k_i^2 + 1)[x_1^2 + (b_i - y_1)^2 - L^2]}}{2(k_i^2 + 1)}. \quad (9)$$

Наконец, для корректного отображения координат цели в полярной системе координат перейдем от прямоугольных координат к полярным координатам по следующим формулам:

$$R = \arctg\left(\frac{y_2}{x_2}\right), \quad (10)$$

$$\theta = \sqrt{x_2^2 + y_2^2}.$$

Таким образом, данный алгоритм позволяет моделировать равномерное движение цели по ломаной линии. Движение цели, смоделированное по описанному алгоритму, представлено на рисунке 13. Моделирование равномерного движения цели по ломаной линии в программном представлении изображено на рисунке 14.

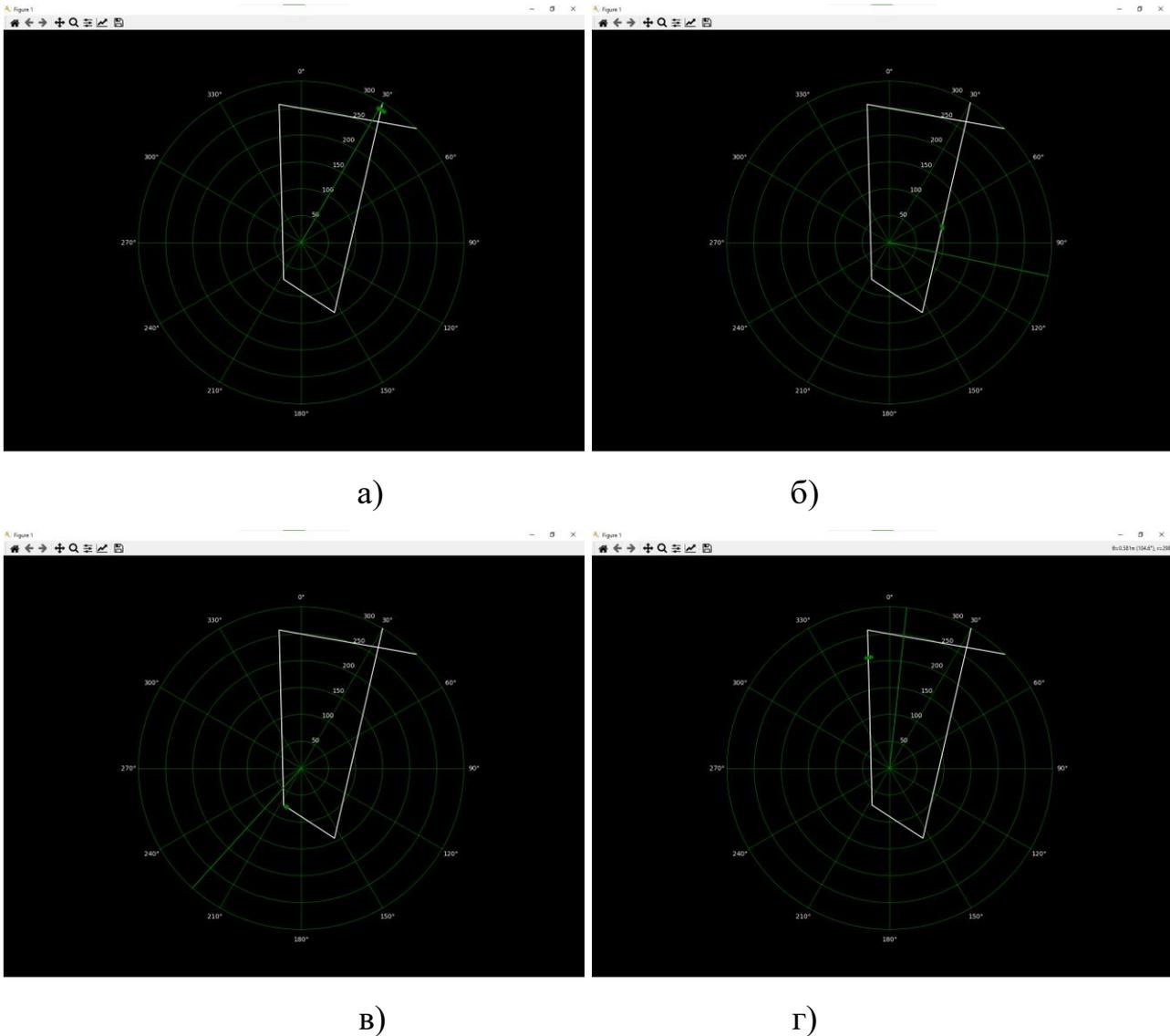


Рис. 13. Моделирование движения цели в произвольном направлении.  
 а) первый оборот РЛС; б) второй оборот РЛС;  
 в) третий оборот РЛС; г) четвёртый оборот РЛС.

```

# Для движения цели по ломанной
if траекторија==2 or траекторија==4:
    if траекторија==2:
        kol_pr=2
    if траекторија==4:
        kol_pr=5
r_t_tr=[random.randint(a: 0, rad_deis) if i>0 and
        i<kol_pr-1 else rad_deis for i in range(kol_pr)]# определение точек пересечения прямых
theta_t_tr=[theta_cel if i==0 else random.randint(a: 0, b: 200)
            * np.pi / 100 for i in range(kol_pr)]
x_t_tr=r_t_tr*np.cos(theta_t_tr)
y_t_tr=r_t_tr*np.sin(theta_t_tr)
k_otr_tr=[(y_t_tr[i+1]-y_t_tr[i])/(x_t_tr[i+1]-x_t_tr[i]) for i in range(kol_pr-1)]
b_otr_tr=[(y_t_tr[i]-k_otr_tr[i]*x_t_tr[i]) for i in range(kol_pr-1)]
D = ((2 * (x - k_otr_tr[q] * (b_otr_tr[q] - y))) ** 2
      - 4 * (k_otr_tr[q] ** 2 + 1) * (x ** 2 + (b_otr_tr[q] - y) ** 2 - skor ** 2))
if x_t_tr[q] < x_t_tr[q + 1]:
    x = (2 * (x - k_otr_tr[q] * (b_otr_tr[q] - y)) +
          D ** 0.5) / (2 * (k_otr_tr[q] ** 2 + 1))
    if x >= x_t_tr[q + 1]:
        x = x_t_tr[q + 1]
        q = q + 1
else:
    x = (2 * (x - k_otr_tr[q] * (b_otr_tr[q] - y)) - D ** 0.5) / (2 * (k_otr_tr[q] ** 2 + 1))
    if x <= x_t_tr[q + 1]:
        x = x_t_tr[q + 1]
        q = q + 1
y = k_otr_tr[q] * x + b_otr_tr[q] # координата точки по оси x
theta_cel = np.arctan(y / x) + np.pi # угол точки траектории
r_cel = ((x**2+y**2)**0.5 / np.cos(theta_cel) # радиус точки траектории

```

Рис. 14. Программное представление моделирования равномерного движения цели по ломаной линии.

### 3. Сравнение моделей в среде MATLAB и в среде Python

При сравнении двух моделей в качестве оценочного параметра было выбрано время, необходимое модели для моделирования одного оборота луча РЛС в разных режимах работы. Были выбраны следующие режимы работы:

- 1) нахождение цели на месте при беспомеховой радиолокационной обстановке;
- 2) движение цели в направлении на РЛС при беспомеховой радиолокационной обстановке;
- 3) движение цели в направлении случайной прямой при беспомеховой радиолокационной обстановке;
- 4) движение цели в направлении случайной прямой при наличии облачности;

5) движение цели в направлении случайной прямой при наличии ложных целей;

6) движение цели в направлении случайной прямой при наличии синхронных импульсных помех;

7) движение цели в направлении случайной прямой при наличии засвеченного от активных помех сектора;

8) движение цели в направлении случайной прямой при наличии всех перечисленных видов помех.

Результаты сравнения двух моделей представлены на графике (рисунок 15). Из графика видно, что новая модель работает заметно быстрее предыдущей и меньше зависит от количества элементов на индикаторе, т.е. от сложности радиолокационной обстановки. Следовательно, можно сделать вывод, что новая модель работает эффективнее предыдущей.

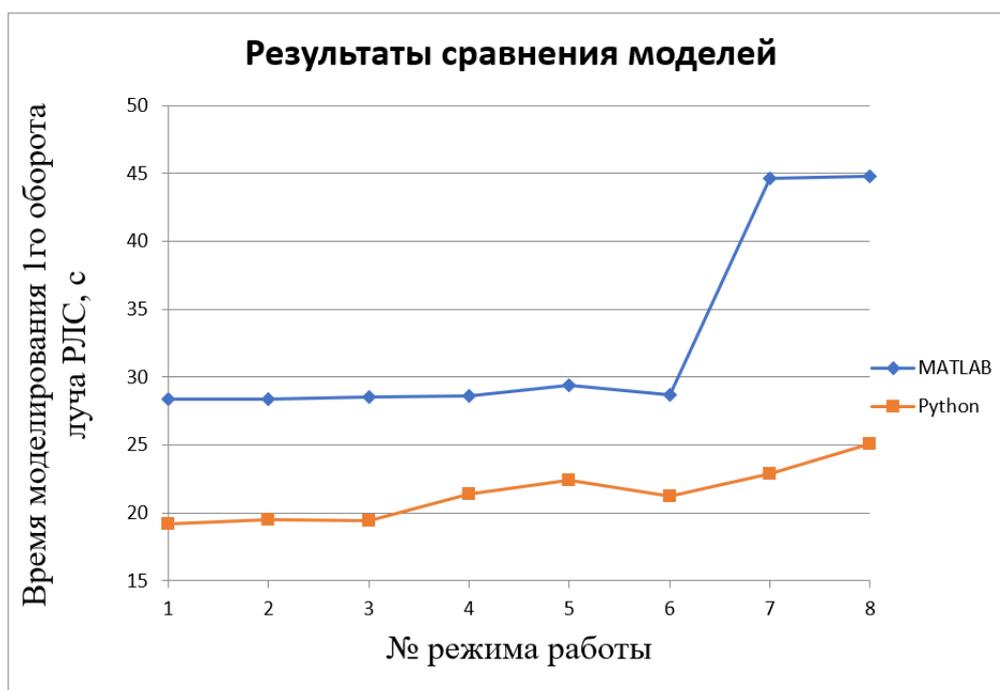


Рис. 15. Результаты сравнения двух моделей.

## Заключение

Разработанная в филиале «Взлёт» МАИ в г. Ахтубинске компьютерная модель индикатора РЛС не требует дополнительного программного обеспечения для работы и может беспрепятственно использоваться в обучении студентов

и операторов РЛС. Новая версия модели не утратила функции прежней модели, и, кроме того, работает заметно эффективнее. Также, разработанная модель, обладает потенциалом дальнейшего развития в области приближения к реальной радиолокационной обстановке индикатора РЛС и использования тактико-технических характеристик реально действующих РЛС.

### Литература

1. Белоруцкий Р. Ю. Оценка искажений отклика приемника РЛС на сигнал, формируемый с помощью двухточечной модели // Журнал радиоэлектроники. – 2012. – №. 1. – С. 4-4.
2. Пинчук А. Н. Модель пассивной помехи при радиолокации низковысотных целей над морской поверхностью // Журнал радиоэлектроники. – 2015. – №. 3. – С. 17-17.
3. Коновской А. С. Моделирование работы индикатора радиолокационной станции кругового обзора / А. С. Коновской, С. А. Рутковский, С. В. Нестеров // Прикаспийский журнал: управление и высокие технологии. – 2024. – № 1(65). – С. 106-115.
4. The MathWorks (Matlab) приостановил деятельность в России с 05 марта 2022 г. [Электрон. ресурс]. (<https://всезапомним.рф/company/704-the-mathworks-matlab.html>) (дата обращения 01.02.2024 г.).
5. Нестеров С. В. Анализ обстановки, отображаемой индикатором кругового обзора радиолокационной станции. – 2013.

#### Для цитирования:

Нестеров С.В., Рутковский С.А. Адаптация модели индикатора кругового обзора радиолокационной станции. // Журнал радиоэлектроники. – 2025. – №. 8. <https://doi.org/10.30898/1684-1719.2025.8.8>