НИУ «МЭИ»
Институт Радиотехники и Электроники им. В. А. Котельникова
Кафедра Радиотехнических Систем

Контрольная работа №1
по курсу «Методы оптимального приёма сигналов в аппаратуре потребителей СРНС»

Выполнила: Малафеева Д. Д.
Группа: ЭР-12м-19

Москва
2020

Дано:
Выборка сигналов, всего M=2048 отсчетов:

$$y_1 = A_1 \cos(\omega kT + \varphi_0) + n_{1,k},$$
$$y_2 = A_1 \sin(\omega kT + \varphi_0) + n_{2,k},$$
$$y_3 = A_2 \cos(\omega kT + \varphi_0 + \Delta\varphi) + n_{3,k},$$
$$y_4 = A_2 \sin(\omega kT + \varphi_0 + \Delta\varphi) + n_{4,k},$$
$$k \in [0, 2047],$$
$$T = 1/f_s,$$
$$f_s = 47.5 МГц.$$

$n_{(1...4),k}$ - независимые некоррелированные по времени ДБГШ с СКО $\sigma_n = 10$.
Параметры сигналов $A_1, A_2, \omega, \varphi_0, \Delta\varphi$ неизвестны, но постоянны на интервале наблюдения.

Найти:
$\Delta\varphi$, дисперсию ошибки для полученной оценки $D_{\Delta\varphi}$.

-

Указания:

- В решении необходимо использовать метод максимального правдоподобия, применять итеративный алгоритм оценивания с помощью дискриминаторов.
- Неинформативные параметры (амплитуда, частота, начальная фаза) считаются информативными и тоже оцениваются:
$$\boldsymbol{\lambda} = \left| A_1 \quad A_2 \quad \omega \quad \varphi_0 \quad \Delta\varphi \right|^T.$$
- Вектор наблюдений:
$$\mathbf{y}_k = \left| y_{1,k} \quad y_{2,k} \quad y_{3,k} \quad y_{4,k} \right|^T = \mathbf{S}_k(\boldsymbol{\lambda}) + \mathbf{n}_k.$$
- Отношение правдоподобия для векторных наблюдений в дискретном времени:
$$\rho\left(\mathbf{Y}_0^M\right) = \exp\left\{ \sum_{k=1}^{M} \mathbf{S}_k^T(\lambda) \mathbf{D}_n^{-1} \left(\mathbf{y}_k - \frac{1}{2} \mathbf{S}_k(\boldsymbol{\lambda}) \right) \right\},$$

где $\mathbf{D}_n = \sigma_n^2 \cdot \mathbf{I}$ - матрица дисперсий шумов наблюдений.

Решение:

Теория

1. Запишем функцию правдоподобия:

$$\ln\left(\rho(\boldsymbol{\lambda})\right) = \sum_{k=1}^{M} \mathbf{S}_k^T(\lambda)\cdot \mathbf{D}_n^{-1}\left(\mathbf{y}_k - \frac{1}{2}\mathbf{S}_k(\boldsymbol{\lambda})\right) = \\ = \sum_{k=1}^{M}\left(\mathbf{S}_k^T(\lambda)\cdot \mathbf{D}_n^{-1}\cdot \mathbf{y}_k - \frac{1}{2}\mathbf{S}_k^T(\boldsymbol{\lambda})\cdot \mathbf{D}_n^{-1}\cdot \mathbf{S}_k(\boldsymbol{\lambda})\right) \tag{0.1}$$

2. Теперь распишем $\mathbf{S}_k^T(\boldsymbol{\lambda})\cdot \mathbf{D}_n^{-1}\cdot \mathbf{S}_k(\boldsymbol{\lambda})$:

$$\mathbf{S}_k^T(\boldsymbol{\lambda}) = \begin{vmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{vmatrix}^T =$$

$$= \begin{vmatrix} A_1\cos\left(\omega kT + \varphi_0\right) & A_1\sin\left(\omega kT + \varphi_0\right) & A_2\cos\left(\omega kT + \varphi_0 + \Delta\varphi\right) & A_2\sin\left(\omega kT + \varphi_0 + \Delta\varphi\right) \end{vmatrix}$$

(0.2)

$$\mathbf{D}_n^{-1} = \begin{bmatrix} \frac{1}{\sigma_n^2} & 0 & 0 & 0 \\ 0 & \frac{1}{\sigma_n^2} & 0 & 0 \\ 0 & 0 & \frac{1}{\sigma_n^2} & 0 \\ 0 & 0 & 0 & \frac{1}{\sigma_n^2} \end{bmatrix} \tag{0.3}$$

$$\mathbf{S}_k^T(\boldsymbol{\lambda})\cdot\mathbf{D}_n^{-1}=$$

$$=\begin{vmatrix} S_1 & S_2 & S_3 & S_4 \end{vmatrix}\cdot\begin{bmatrix} \frac{1}{\sigma_n^2} & 0 & 0 & 0 \\ 0 & \frac{1}{\sigma_n^2} & 0 & 0 \\ 0 & 0 & \frac{1}{\sigma_n^2} & 0 \\ 0 & 0 & 0 & \frac{1}{\sigma_n^2} \end{bmatrix}=$$

$$=\begin{vmatrix} \frac{1}{\sigma_n^2}\cdot S_1 & \frac{1}{\sigma_n^2}\cdot S_2 & \frac{1}{\sigma_n^2}\cdot S_3 & \frac{1}{\sigma_n^2}\cdot S_4 \end{vmatrix} \qquad (0.4)$$

Тогда $\mathbf{S}_k^T(\boldsymbol{\lambda})\cdot\mathbf{D}_n^{-1}\cdot\mathbf{S}_k(\boldsymbol{\lambda})$:

$$\mathbf{S}_k^T(\boldsymbol{\lambda})\cdot\mathbf{D}_n^{-1}\cdot\mathbf{S}_k(\boldsymbol{\lambda})=$$

$$=\begin{vmatrix} \frac{1}{\sigma_n^2}\cdot S_1 & \frac{1}{\sigma_n^2}\cdot S_2 & \frac{1}{\sigma_n^2}\cdot S_3 & \frac{1}{\sigma_n^2}\cdot S_4 \end{vmatrix}\cdot\begin{vmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{vmatrix}=$$

$$=\frac{1}{\sigma_n^2}\cdot S_1\cdot S_1+\frac{1}{\sigma_n^2}\cdot S_2\cdot S_2+\frac{1}{\sigma_n^2}\cdot S_3\cdot S_3+\frac{1}{\sigma_n^2}\cdot S_4\cdot S_4=$$

$$=\frac{1}{\sigma_n^2}\left(S_1^2+S_2^2+S_3^2+S_4^2\right)=$$

$$=\frac{1}{\sigma_n^2}[A_1^2\cos^2\left(\omega kT+\varphi_0\right)+A_1^2\sin^2\left(\omega kT+\varphi_0\right)+$$

$$+A_2^2\cos^2\left(\omega kT+\varphi_0+\Delta\varphi\right)+A_2^2\sin^2\left(\omega kT+\varphi_0+\Delta\varphi\right)]=$$

$$=\frac{1}{\sigma_n^2}\left(A_1^2+A_2^2\right). \qquad (0.5)$$

3. Вернемся к выражению 1.1 и запишем его с учетом приведенных преобразований:

$$
\begin{aligned}
\ln\left(\rho(\boldsymbol{\lambda})\right) &= \sum_{k=1}^{M} \mathbf{S}_k^T(\lambda)\cdot \mathbf{D}_n^{-1}\left(\mathbf{y}_k - \frac{1}{2}\mathbf{S}_k(\boldsymbol{\lambda})\right) = \\
&= \sum_{k=1}^{M}\left(\mathbf{S}_k^T(\lambda)\cdot \mathbf{D}_n^{-1}\cdot \mathbf{y}_k - \frac{1}{2}\mathbf{S}_k^T(\boldsymbol{\lambda})\cdot \mathbf{D}_n^{-1}\cdot \mathbf{S}_k(\boldsymbol{\lambda})\right) = \\
&= \sum_{k=1}^{M}\left(\left|\frac{1}{\sigma_n^2}\cdot S_1 \quad \frac{1}{\sigma_n^2}\cdot S_2 \quad \frac{1}{\sigma_n^2}\cdot S_3 \quad \frac{1}{\sigma_n^2}\cdot S_4\right|\cdot \mathbf{y}_k - \frac{1}{2}\cdot\frac{1}{\sigma_n^2}\left(A_1^2 + A_2^2\right)\right) = \\
&= \frac{1}{\sigma_n^2}\left[\cdot\sum_{k=1}^{M}\left(S_1\cdot y_1 + S_2\cdot y_2 + S_3\cdot y_3 + S_4\cdot y_4\right) - \frac{M}{2}\left(A_1^2 + A_2^2\right)\right]
\end{aligned}
\quad (0.6)
$$

4. Далее рассчитаем все производные функции правдоподобия по составляющим вектора оцениваемых параметров:

$$
\frac{\partial \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_1} = \frac{1}{\sigma_n^2}\left[\sum_{k=1}^{M}\left(\cos\left(\omega kT + \varphi_0\right)y_{1,k} + \sin\left(\omega kT + \varphi_0\right)y_{2,k}\right) - M\cdot A_1\right],
$$

(0.7)

$$
\frac{\partial \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_2} = \frac{1}{\sigma_n^2}\left[\sum_{k=1}^{M}\left(\cos\left(\omega kT + \varphi_0 + \Delta\varphi\right)y_{3,k} + \sin\left(\omega kT + \varphi_0 + \Delta\varphi\right)y_{4,k}\right) - M\cdot A_2\right],
$$

(0.8)

$$
\begin{aligned}
\frac{\partial \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \omega} = \frac{1}{\sigma_n^2}[\sum_{k=1}^{M}(&-A_1\sin\left(\omega kT + \varphi_0\right)y_{1,k}\cdot kT + A_1\cos\left(\omega kT + \varphi_0\right)y_{2,k}\cdot kT - \\
&-A_2\sin\left(\omega kT + \varphi_0 + \Delta\varphi\right)y_{3,k}\cdot kT + A_2\cos\left(\omega kT + \varphi_0\right)y_{4,k}\cdot kT)],
\end{aligned}
$$

(0.9)

$$
\begin{aligned}
\frac{\partial \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \varphi_0} = \frac{1}{\sigma_n^2}[\sum_{k=1}^{M}(&-A_1\sin\left(\omega kT + \varphi_0\right)y_{1,k} + A_1\cos\left(\omega kT + \varphi_0\right)y_{2,k} - \\
&-A_2\sin\left(\omega kT + \varphi_0 + \Delta\varphi\right)y_{3,k} + A_2\cos\left(\omega kT + \varphi_0\right)y_{4,k})],
\end{aligned}
$$

(0.10)

$$
\frac{\partial \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \Delta\varphi} = \frac{1}{\sigma_n^2}[\sum_{k=1}^{M}(-A_2\sin\left(\omega kT + \varphi_0 + \Delta\varphi\right)y_{3,k} + A_2\cos\left(\omega kT + \varphi_0\right)y_{4,k})],
$$

(0.11)

5. Теперь найдем все вторые и смешанные производные функции правдоподобия:

- A_1:

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial A_1^{\,2}} = -\frac{M}{\sigma_n^2}, \tag{0.12}$$

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial A_1 \partial A_2} = 0, \tag{0.13}$$

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial A_1 \partial \omega} = \frac{1}{\sigma_n^2}\left[\sum_{k=1}^{M}\left(-\sin\left(\omega kT+\varphi_0\right)y_{1,k}\cdot kT+\cos\left(\omega kT+\varphi_0\right)y_{2,k}\cdot kT\right)\right], \tag{0.14}$$

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial A_1 \partial \varphi_0} = \frac{1}{\sigma_n^2}\left[\sum_{k=1}^{M}\left(-\sin\left(\omega kT+\varphi_0\right)y_{1,k}+\cos\left(\omega kT+\varphi_0\right)y_{2,k}\right)\right], \tag{0.15}$$

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial A_1 \partial \Delta\varphi} = 0. \tag{0.16}$$

- A_2:

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial A_2^{\,2}} = -\frac{M}{\sigma_n^2}, \tag{0.17}$$

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial A_2 \partial \omega} = \frac{1}{\sigma_n^2}\left[\sum_{k=1}^{M}\left(-\sin\left(\omega kT+\varphi_0+\Delta\varphi\right)y_{3,k}\cdot kT+\cos\left(\omega kT+\varphi_0+\Delta\varphi\right)y_{4,k}\cdot kT\right)\right], \tag{0.18}$$

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial A_2 \partial \varphi_0} = \frac{1}{\sigma_n^2}\left[\sum_{k=1}^{M}\left(-\sin\left(\omega kT+\varphi_0+\Delta\varphi\right)y_{3,k}+\cos\left(\omega kT+\varphi_0+\Delta\varphi\right)y_{4,k}\right)\right], \tag{0.19}$$

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial A_2 \partial \Delta\varphi} = \frac{1}{\sigma_n^2}\left[\sum_{k=1}^{M}\left(-\sin\left(\omega kT+\varphi_0+\Delta\varphi\right)y_{3,k}+\cos\left(\omega kT+\varphi_0+\Delta\varphi\right)y_{4,k}\right)\right]. \tag{0.20}$$

- ω:

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial \omega^2} = \frac{1}{\sigma_n^2}[\sum_{k=1}^{M}(-A_1 \cos\left(\omega kT + \varphi_0\right) y_{1,k} \cdot \left(kT\right)^2 - A_1 \sin\left(\omega kT + \varphi_0\right) y_{2,k} \cdot \left(kT\right)^2 -$$
$$-A_2 \cos\left(\omega kT + \varphi_0 + \Delta\varphi\right) y_{3,k} \cdot \left(kT\right)^2 - A_2 \sin\left(\omega kT + \varphi_0\right) y_{4,k} \cdot \left(kT\right)^2)],$$
(0.21)

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial \omega \partial \varphi_0} = \frac{1}{\sigma_n^2}[\sum_{k=1}^{M}(-A_1 \cos\left(\omega kT + \varphi_0\right) y_{1,k} \cdot kT - A_1 \sin\left(\omega kT + \varphi_0\right) y_{2,k} \cdot kT -$$
$$-A_2 \cos\left(\omega kT + \varphi_0 + \Delta\varphi\right) y_{3,k} \cdot kT - A_2 \sin\left(\omega kT + \varphi_0\right) y_{4,k} \cdot kT)],$$
(0.22)

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial \omega \partial \Delta\varphi} = \frac{1}{\sigma_n^2}[-A_2 \cos\left(\omega kT + \varphi_0 + \Delta\varphi\right) y_{3,k} \cdot kT - A_2 \sin\left(\omega kT + \varphi_0\right) y_{4,k} \cdot kT)],$$
(0.23)

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial {\varphi_0}^2} = \frac{1}{\sigma_n^2}[\sum_{k=1}^{M}(-A_1 \cos\left(\omega kT + \varphi_0\right) y_{1,k} - A_1 \sin\left(\omega kT + \varphi_0\right) y_{2,k} -$$
$$-A_2 \cos\left(\omega kT + \varphi_0 + \Delta\varphi\right) y_{3,k} - A_2 \sin\left(\omega kT + \varphi_0\right) y_{4,k})],$$
(0.24)

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial \varphi_0 \partial \Delta\varphi} = \frac{1}{\sigma_n^2}[\sum_{k=1}^{M}(- A_2 \cos\left(\omega kT + \varphi_0 + \Delta\varphi\right) y_{3,k} - A_2 \sin\left(\omega kT + \varphi_0\right) y_{4,k})].$$
(0.25)

- $\Delta\varphi$:

$$\frac{\partial^2 \ln\left(\rho\left(\boldsymbol{\lambda}\right)\right)}{\partial \Delta\varphi^2} = \frac{1}{\sigma_n^2}[\sum_{k=1}^{M}(- A_2 \cos\left(\omega kT + \varphi_0 + \Delta\varphi\right) y_{3,k} - A_2 \sin\left(\omega kT + \varphi_0\right) y_{4,k})].$$
(0.26)

6. Далее все рассчитанные производные нужно собрать в вектор (для первых производных) и матрицу (для вторых и смешанных производных) и применить алгоритм оценивания параметров сигналов с помощью дискриминаторов:

$$\lambda^{(k+1)} = \lambda^{(k)} - \frac{\partial \ln\left[\rho\left(\lambda^{(k)}\right)\right]}{\partial \lambda}\left[\frac{\partial^2 \ln\left[\rho\left(\lambda^{(k)}\right)\right]}{\partial \lambda^2}\right]^{-1}, \qquad (0.27)$$

Здесь

$$\frac{\partial \ln\left[\rho\left(\boldsymbol{\lambda}^{(k)}\right)\right]}{\partial \boldsymbol{\lambda}} =$$

$$= \left| \frac{\partial \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_1} \quad \frac{\partial \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_2} \quad \frac{\partial \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \omega} \quad \frac{\partial \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \varphi_0} \quad \frac{\partial \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \Delta\varphi} \right|,$$

(0.28)

$$\left[\frac{\partial^2 \ln\left[\rho\left(\lambda^{(k)}\right)\right]}{\partial \lambda^2}\right]^{-1} =$$

$$= \begin{bmatrix}
\frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_1^2} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_1 \partial A_2} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_1 \partial \omega} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_1 \partial \varphi_0} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_1 \partial \Delta\varphi} \\
\frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_1 \partial A_2} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_2^2} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_2 \partial \omega} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_2 \partial \varphi_0} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_2 \partial \Delta\varphi} \\
\frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_1 \partial \omega} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_2 \partial \omega} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \omega^2} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \omega \partial \varphi_0} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \omega \partial \Delta\varphi} \\
\frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_1 \partial \varphi_0} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_2 \partial \varphi_0} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \omega \partial \varphi_0} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \varphi_0^2} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \varphi_0 \partial \Delta\varphi} \\
\frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_1 \partial \Delta\varphi} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial A_2 \partial \Delta\varphi} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \omega \partial \Delta\varphi} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \varphi_0 \partial \Delta\varphi} & \frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial \Delta\varphi^2}
\end{bmatrix}^{-1}.$$

(0.29)

Расчет производится с помощью метода простой итерации. Критерий окончания:

$$\left|\Delta\varphi^{(k)} - \Delta\varphi^{(k-1)}\right| \le 10^{-8}. \qquad (0.30)$$

7. При условии, что ОСШ достаточно большое дисперсию ошибки оценивания параметра $\Delta\varphi$ - $D_{\Delta\varphi}$ можно найти по формуле:

$$D_{\Delta\hat{\varphi}} = J_{5,5}^{-1}, \qquad (0.31)$$

где $J_{i,j}$ - это элемент матрицы Фишера, рассчитываемый по формуле:

$$J_{i,j} = -\mathrm{M}\left[\frac{\partial^2 \ln\left(\rho(\lambda)\right)}{\partial\lambda_i \partial\lambda_j}\right],$$

(0.32)

здесь $\mathrm{M}[*]$- функция вычисления мат. ожидания.

Тогда:

$$D_{\Delta\varphi} = J_{5,5}^{-1} = -\left[\frac{\partial^2 \ln\left(\rho(\boldsymbol{\lambda})\right)}{\partial\Delta\varphi^2}\right]^{-1}. \qquad (0.33)$$

Математическое моделирование

1. Начальные данные для алгоритма оценивания параметров с помощью дискриминаторов

Расчёты проведены на языке программирования Python. Построим выборку заданных реализаций на входе приёмника:

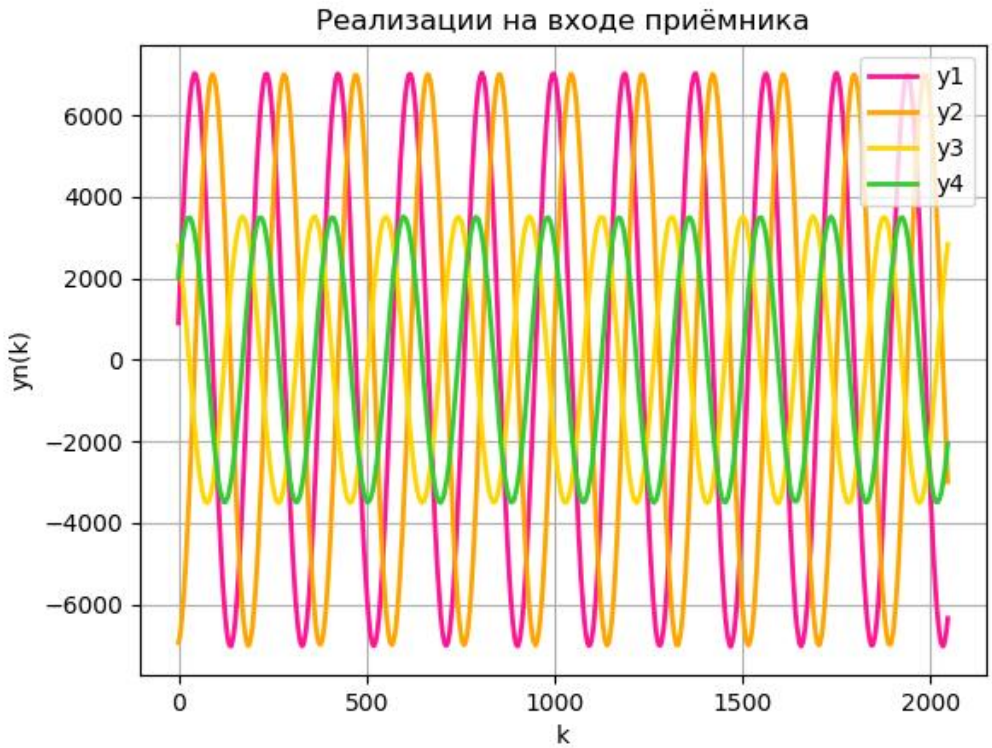

Рис. 1 Реализации на входе приёмника

Необходимо задать начальное значение вектора оцениваемых параметров:

$$\boldsymbol{\lambda}_0 = \begin{vmatrix} A_{1,0} & A_{2,0} & \omega_0 & \varphi_{0,0} & \Delta\varphi_0 \end{vmatrix}^T = \\ = \begin{vmatrix} 7030 & 3500 & 5 \cdot 10^5 & rad(-60^\circ) & rad(-120^\circ) \end{vmatrix}^T. \tag{0.34}$$

Все параметры выбраны, вероятно, далекими от начальных значений параметров. Однако, мне кажется, это должно повлиять только на количество итераций, за которые оценка параметров достигнет заданной точности.

Результаты

(пока их нет)

Листинг программы

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Mon Oct 19 02:18:01 2020

@author: daryamalafeeva
"""
import numpy as np
import codecs
import matplotlib.pyplot as plt
import math
from numpy.linalg import inv

"""--------------------------Обработка файла------------------------------"""
# инициализация списков
y1_list = []
y2_list = []
y3_list = []
y4_list = []

with codecs.open('Input_Y0toT.txt', "r", encoding='utf-8', errors='ignore') as log:
    for line in log:
        str_massive = line.split()
        y1         = int(str_massive[0])
        y1_list.append(y1)
        y2         = int(str_massive[1])
        y2_list.append(y2)
        y3         = int(str_massive[2])
        y3_list.append(y3)
        y4         = int(str_massive[3])
        y4_list.append(y4)

figure_1 = plt.figure(1)
plt.plot(range(0, len(y1_list)), y1_list, color = 'deeppink', linewidth = 2)
plt.plot(range(0, len(y2_list)), y2_list, color = 'orange', linewidth = 2)
plt.plot(range(0, len(y3_list)), y3_list, color = 'gold', linewidth = 2)
plt.plot(range(0, len(y4_list)), y4_list, color = 'limegreen', linewidth = 2)
plt.xlabel('k')
plt.ylabel('yn(k)')
plt.legend(['y1','y2','y3', 'y4'])
plt.title('Реализации на входе приёмника')
plt.grid()
plt.show()

"""-------------------------Параметры моделирования-----------------------"""
f_s    = 47.5*1e6
T      = 1/f_s
M      = 2048
sigma_n = 10

"""--------------------Начальные параметры алгоритма----------------------"""
# начальные значения параметров вектора lam_array
A1       = 7030
A2       = 3500
f        = 5*1e5
```

```
w        = 2 * math.pi * f
phi_0     = math.radians(-60)
delta_phi = math.radians(-120)
lam_array = np.array([A1, A2, w, phi_0, delta_phi])

S1_list = []
S2_list = []
S3_list = []
S4_list = []

"""--------------------Алгоритм оценивания параметров---------------------"""
# начальное значение для метода простой итерацииs
delta_phi_old = 0.1 * delta_phi

for k in range(1, M, 1):
    while abs(delta_phi - delta_phi_old) > 1e-8:
        # первые производные функции правдоподобия
        d1_dA1 = 1/(sigma_n**2) *\
                (sum(math.cos(w * k * T + phi_0) * np.array(y1_list)+\
                    math.sin(w * k * T + phi_0) * np.array(y2_list)) - M * A1)

        d1_dA2 = 1/(sigma_n**2) *\
                (sum(math.cos(w * k * T + phi_0 + delta_phi) * np.array(y3_list)+\
                    math.sin(w * k * T + phi_0 + delta_phi) * np.array(y4_list)) - M * A2)

        d1_dw  = 1/(sigma_n**2) *\
                (sum(-A1 * math.sin(w * k * T + phi_0) * k * T * np.array(y1_list)+\
                      A1 * math.cos(w * k * T + phi_0) * k * T * np.array(y2_list)-\
                      A2 * math.sin(w * k * T + phi_0 + delta_phi) * k * T * np.array(y3_list)+\
                      A2 * math.cos(w * k * T + phi_0 + delta_phi) * k * T * np.array(y4_list)))

        d1_dphi_0 = 1/(sigma_n**2) *\
                    (sum(-A1 * math.sin(w * k * T + phi_0) * np.array(y1_list)+\
                          A1 * math.cos(w * k * T + phi_0) * np.array(y2_list)-\
                          A2 * math.sin(w * k * T + phi_0 + delta_phi) * np.array(y3_list)+\
                          A2 * math.cos(w * k * T + phi_0 + delta_phi) * np.array(y4_list)))

        d1_ddelta_phi = 1/(sigma_n**2) *\
                        (sum(-A2 * math.sin(w * k * T + phi_0 + delta_phi) * np.array(y3_list)+\
                              A2 * math.cos(w * k * T + phi_0 + delta_phi) * np.array(y4_list)))

        L = np.array([d1_dA1, d1_dA2, d1_dw, d1_dphi_0, d1_ddelta_phi])

        # вторые и смешанные производные функции правдоподобия
        d2_dA1    = (-M)/(sigma_n**2)

        d2_dA1dA2 = 0

        d2_dA1dw  = 1/(sigma_n**2) *\
                    (sum(-math.sin(w * k * T + phi_0) * k * T * np.array(y1_list)+\
                          math.cos(w * k * T + phi_0) * k * T * np.array(y2_list)))

        d2_dA1dphi_0 = 1/(sigma_n**2) *\
                        (sum(-math.sin(w * k * T + phi_0) * np.array(y1_list)+\
                              math.cos(w * k * T + phi_0) * np.array(y2_list)))
```

```
        d2_dA1ddelta_phi = 0

        d2_dA2    = (-M)/(sigma_n**2)

        d2_dA2dw = 1/(sigma_n**2) *\
                (sum(-math.sin(w * k * T + phi_0 + delta_phi) * k * T * np.array(y3_list)+\
                    math.cos(w * k * T + phi_0 + delta_phi) * k * T * np.array(y4_list)))

        d2_A2dphi_0 = 1/(sigma_n**2) *\
                (sum(-math.sin(w * k * T + phi_0 + delta_phi) * np.array(y3_list)+\
                    math.cos(w * k * T + phi_0 + delta_phi) * np.array(y4_list)))

        d2_dA2ddelta_phi = 1/(sigma_n**2) *\
                  (sum(-math.sin(w * k * T + phi_0 + delta_phi) * np.array(y3_list)+\
                      math.cos(w * k * T + phi_0 + delta_phi) * np.array(y4_list)))

        d2_dw = 1/(sigma_n**2) *\
                (sum(-A1 * math.cos(w * k * T + phi_0) * ((k * T)**2) * np.array(y1_list)-\
                    A1 * math.sin(w * k * T + phi_0) * ((k * T)**2) * np.array(y2_list)-\
                    A2 * math.cos(w * k * T + phi_0 + delta_phi) * ((k * T)**2) * np.array(y3_list)-
\
                    A2 * math.sin(w * k * T + phi_0 + delta_phi) * ((k * T)**2) * np.array(y4_list)))

        d2_dwdphi_0 = 1/(sigma_n**2) *\
                (sum(-A1 * math.cos(w * k * T + phi_0) * k * T * np.array(y1_list)-\
                    A1 * math.sin(w * k * T + phi_0) * k * T * np.array(y2_list)-\
                    A2 * math.cos(w * k * T + phi_0 + delta_phi) * k * T * np.array(y3_list)-\
                    A2 * math.sin(w * k * T + phi_0 + delta_phi) * k * T * np.array(y4_list)))

        d2_dwddelta_phi = 1/(sigma_n**2) *\
                  (sum(-A2 * math.cos(w * k * T + phi_0 + delta_phi) * k * T * np.array(y3_list)-\
                      A2 * math.sin(w * k * T + phi_0 + delta_phi) * k * T * np.array(y4_list)))

        d2_dphi_0 = 1/(sigma_n**2) *\
                (sum(-A1 * math.cos(w * k * T + phi_0) * np.array(y1_list)-\
                    A1 * math.sin(w * k * T + phi_0) * np.array(y2_list)-\
                    A2 * math.cos(w * k * T + phi_0 + delta_phi) * np.array(y3_list)-\
                    A2 * math.sin(w * k * T + phi_0 + delta_phi) * np.array(y4_list)))

        d2_dphi_0ddelta_phi = 1/(sigma_n**2) *\
                (sum(-A2 * math.cos(w * k * T + phi_0 + delta_phi) * np.array(y3_list)-\
                    A2 * math.sin(w * k * T + phi_0 + delta_phi) * np.array(y4_list)))

        d2_ddelta_phi = 1/(sigma_n**2) *\
                (sum(-A2 * math.cos(w * k * T + phi_0 + delta_phi) * np.array(y3_list)-\
                    A2 * math.sin(w * k * T + phi_0 + delta_phi) * np.array(y4_list)))

        H = np.array([[d2_dA1,          d2_dA1dA2,        d2_dA1dw,        d2_dA1dphi_0,
d2_dA2ddelta_phi],
                 [d2_dA1dA2,         d2_dA2,             d2_dA2dw,          d2_A2dphi_0,
d2_dA2ddelta_phi],
                 [d2_dA1dw,          d2_dA2dw,           d2_dw,             d2_dwdphi_0,
d2_dwddelta_phi],
                 [d2_dA1dphi_0,      d2_A2dphi_0,        d2_dwdphi_0,       d2_dphi_0,
d2_dphi_0ddelta_phi],
```

```
                [d2_dA2ddelta_phi,              d2_dA2ddelta_phi,              d2_dwddelta_phi,
d2_dphi_0ddelta_phi, d2_ddelta_phi]])

        # метод дискриминаторов
        lam_array_old = lam_array
        lam_array     = lam_array - np.dot(L,H)

        # обновляем параметры
        A1           = lam_array[0]
        A2           = lam_array[1]
        w            = lam_array[2]
        pho_0        = lam_array[3]
        delta_phi_old = delta_phi
        delta_phi     = lam_array[4]

    # выход из цикла while
    # ошибка оценивания delta_phi
    J = -H
    D_lambda = inv(J)
    D_delta_phi = D_lambda[4,4]

    # сигналы с оценками параметров
    S1 = A1 * math.cos(w * k * T + phi_0)
    S1_list.append(S1)
    S2 = A1 * math.sin(w * k * T + phi_0)
    S2_list.append(S2)
    S3 = A2 * math.cos(w * k * T + phi_0 + delta_phi)
    S3_list.append(S3)
    S4 = A2 * math.sin(w * k * T + phi_0 + delta_phi)
    S4_list.append(S4)

figure_2 = plt.figure(2)
plt.plot(range(0, len(S1_list)), S1_list, color = 'deeppink', linewidth = 2)
plt.plot(range(0, len(S2_list)), S2_list, color = 'orange', linewidth = 2)
plt.plot(range(0, len(S3_list)), S3_list, color = 'gold', linewidth = 2)
plt.plot(range(0, len(S4_list)), S4_list, color = 'limegreen', linewidth = 2)
plt.xlabel('k')
plt.ylabel('Sn(k)')
plt.legend(['S1','S2','S3', 'S4'])
plt.title('Сигналы с оценками параметров ')
plt.grid()
plt.show()
```