

Применение нейронной сети для расчета сопротивления нагрузок с учетом инвариантных свойств соотношения ВХОД—ВЫХОД МНОГОПОЛЮСНИКОВ

ПЕНИН А.А.

ИЭИН им. Д. Гицу, Кишинев, Республика Молдова

Рассматриваются инвариантные свойства соотношения вход–выход многополюсников как модели нестабильной двух- и трехпроводной линии питания силовых нагрузок или резистивных датчиков. Дробно-линейные выражения, характерные для теории электрических цепей, интерпретируются как проективные преобразования в смысле проективной геометрии. Проективные преобразования сохраняют инвариант: сложное отношение или двойная пропорция четырех выборок изменяемого сопротивления и соответствующих значений тока на разных участках цепи. Это позволяет рассчитать сопротивление нагрузок по измеренным выборкам токов на входе цепи и их известным базовым или тестовым значениям, причем параметры цепи явно не используются. В свою очередь, базовые значения нагрузок также могут изменяться или известны с какой-то погрешностью. Чтобы исключить влияние изменяемых параметров для расчета применяются нейронные сети. Для обучения задаётся набор из возможных значений базовых сопротивлений, нагрузок и изменяемого сопротивления самого многополюсника. Соответствующий набор токов на входе образует вектор обучения, а значения нагрузок являются компонентами вектора цели. Численные эксперименты в пакете Deep Learning системы MATLAB с одной и двумя нагрузками показывают точность контрольного расчета для обученной поверхностной нейронной сети на уровне одного-двух процентов.

К л ю ч е в ы е с л о в а: многополюсник, нагрузочная характеристика, проводная линия, проективные преобразования, сложное отношение, нейронная сеть

Рассмотрим линейный многополюсник с определенным одинаковым числом входов и выходов. Подобные цепи могут быть моделью нестабильной двух- и трехпроводной линии питания силовых нагрузок или резистивных датчиков. В случае заданных параметров многополюсника известные соотношения позволяют рассчитать сопротивления нагрузок по измеренным токам на входе. Если параметры цепи неизвестны или меняются, определяют эквивалентные параметры подходящими методами измерений, например как в [1, 2].

С другой стороны, в ряде работ [3, 4] автором рассматриваются инвариантные соотношения вход–выход многополюсников. Этот подход основан на интерпретации характерных для теории электрических цепей дробно-линейных выражений как проективных преобразований (отображение точек, линий и т.д.) в смысле проективной геометрии [3, 4]. Проективные преобразования сохраняют инвариант – сложное отношение (двойная пропорция) четырех выборок (значений) изменяемого сопротивления и соответствующих значений токов на разных участках цепи. В качестве трех выборок используются задаваемые базовые (тестовые) значения нагрузок, а четвертой выборкой является текущее значение каждой нагрузки. Поэтому можно рассчитать сопротивление нагрузок по измеренным выборкам токов на входе цепи и базовым значениям.

Важно отметить, что параметры цепи явно не рассчитываются. Привлекает внимание структура выражения сложного отношения: ошибки измерения токов взаимно сокращаются [5–7].

В свою очередь, базовые значения нагрузок могут изменяться или быть известны с какой-то погрешностью, что исключает непосредственное определение сопротивления нагрузок по измеренным выборкам токов. В настоящей работе рассматривается применение аппарата нейронных сетей, которые снимают эту проблему. Нейронные сети являются универсальными аппроксиматорами функции нескольких переменных [8–10]. Важное свойство сети состоит в способности к обучению и обобщению входных данных, поэтому она способна выполнять разные функции, такие как аппроксимация и интерполяция, прогнозирование, фильтрация, распознавание. Параллельная обработка сигналов всеми нейронами обеспечивает высокое быстродействие, что является важным для обработки сигналов в реальном времени.

Для обучающей базы данных задаётся набор из возможных значений базовых сопротивлений, нагрузок и изменяемого сопротивления многополюсника. Соответствующий набор токов на входе образует вектор обучения, а значения нагрузок являются компонентами вектора цели. Численные эксперименты в пакете Deep

Learning системы *MATLAB* [11] с одной и двумя нагрузками для обученной поверхностной сети показывают точность контрольного расчета на уровне одного-двух процентов. Чтобы выделить содержательную сторону подхода и упростить изложение, рассмотрены цепи постоянного тока.

Проективные преобразования вход–выход многополюсника. Приведем необходимые сведения из интерпретаций уравнений и графических характеристик ряда цепей как проективных преобразований и их применение для вычисления нагрузок по измеренным токам на входах.

Цепь с одной нагрузкой. Проективные координаты точки на прямой линии. Уравнение нагрузочной или вольт-амперной характеристики (ВАХ) активного двухполюсника *A* на рис. 1

$$I_H = \frac{U_0}{R_i} - \frac{U_H}{R_i} = I_H^{SC} - \frac{U_H}{R_i} \quad (1)$$

представляет прямую линию. В свою очередь, уравнение нагрузки

$$I_H = \frac{1}{R_H} U_H \quad (2)$$

определяет пучок прямых линий с параметром R_H и центром в точке 0. Этот пучок линий пересекают ВАХ, поэтому можно проградуировать ВАХ в значениях сопротивления нагрузки R_H .

Величина R_H определяет неоднородную координату рабочей точки на линии ВАХ. В свою очередь, такие величины как U_H , I_H являются однородными координатами, отношение которых определяет значение R_H .

Бесконечно удаленная точка $R_H = \infty$ соответствует конечной точке $U_H = U_0$.

Рассмотрим напряжение нагрузки. Уравнение $U_H(R_H)$ представляет характерное дробно-линейное выражение:

$$U_H = U_0 \frac{R_H}{R_i + R_H}. \quad (3)$$

Изложенное выше даёт основание считать отображение точек прямой линии R_H на прямую линию U_H как одномерное проективное преобразование [12–14]. В частности, проективное преобразование задаётся центром проекции *S* или тремя парами соответственных базовых точек, как условно показано на рис. 2. Пусть это будет какое-то максимальное значение R_H^{OC} , минимальное R_H^{SC} и опорное R_H^{REF} . В частности,

$$R_H^{OC} = \infty; R_H^{SC} = 0; R_H^{REF} = R_i. \quad (4)$$

Для текущих параметров нагрузки R_H^1 , U_H^1 , I_H^1 сложное отношение

$$m_H^1 = (R_H^{SC} R_H^1 R_H^{REF} R_H^{OC}) = \frac{R_H^1 - R_H^{SC}}{R_H^1 - R_H^{OC}} \div \frac{R_H^{REF} - R_H^{SC}}{R_H^{REF} - R_H^{OC}} = \\ = (U_H^{SC} U_H^1 U_H^{REF} U_H^{OC}) = (I_H^{SC} I_H^1 I_H^{REF} I_H^{OC}). \quad (5)$$

Для частного случая базовых точек (4) сложное отношение

$$m_H^1 = (0 R_H^1 R_H^{REF} \infty) = \frac{R_H^1}{R_H^{REF}} = \frac{U_H^1}{U_0 - U_H^1} = \frac{I_H^{SC} - I_H^1}{I_H^1}.$$

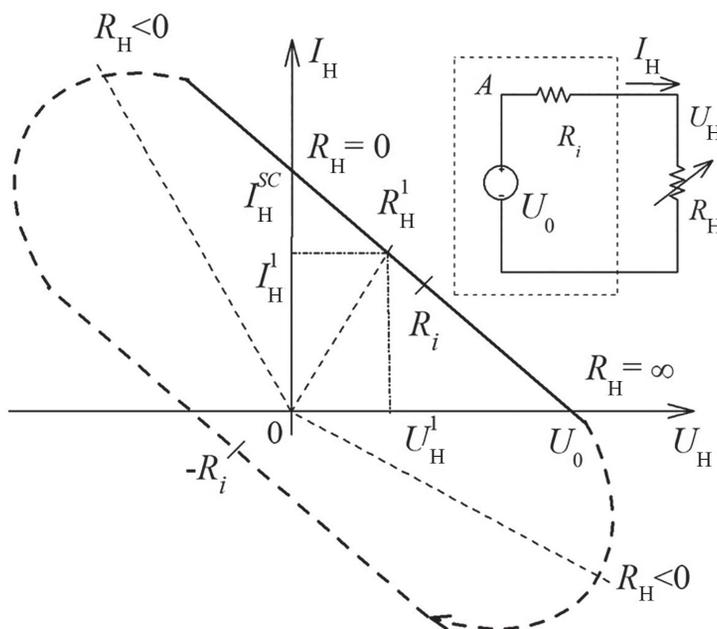


Рис. 1. Схема и нагрузочная прямая линия активного двухполюсника *A*

Fig. 1. Circuit and the load straight line of active two-pole *A*

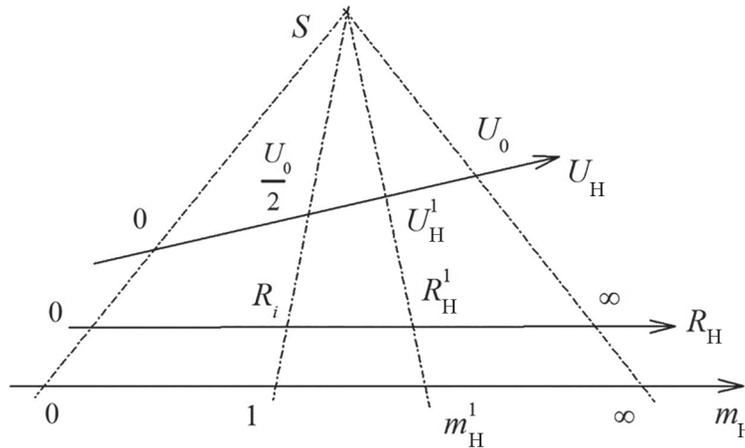


Рис. 2. Проективное преобразование $R_H \rightarrow U_H$
 Fig. 2. Projective transformation of $R_H \rightarrow U_H$

Таким образом, проективная координата текущей рабочей точки ВАХ в безразмерном или относительном виде задается величиной сложного отношения m_H^1 .

Инварианты вход–выход четырехполюсника. Рассмотрим четырехполюсник 4П на рис. 3. Здесь и далее будем использовать проводимости элементов цепи.

Система уравнений имеет вид:

$$\begin{bmatrix} I_1 \\ I_0 \end{bmatrix} = \begin{bmatrix} -Y_{11} & Y_{10} \\ -Y_{10} & Y_{00} \end{bmatrix} \begin{bmatrix} U_1 \\ U_0 \end{bmatrix}; \quad (6)$$

$$Y_{11} = y_{10} + y_1; \quad Y_{00} = y_{10} + y_0; \quad Y_{10} = y_{10}.$$

При изменении проводимости нагрузки Y_H имеем ВАХ на выходе и входе. Пучки прямых линий вида (2)

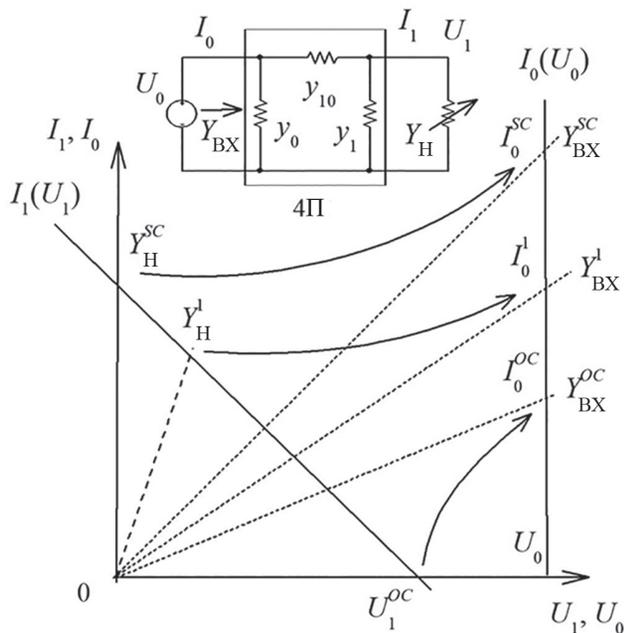


Рис. 3. Схема и отображение нагрузочных характеристик выход–вход четырехполюсника 4П

Fig. 3. Circuit and mapping of load characteristics of 4P two-port

с параметрами Y_H, Y_{BX} соответствуют этим характеристикам. Также можно проградуировать ВАХ в значениях проводимостей. Непропорциональность полученных градуировок соответствует отображению $Y_H \rightarrow Y_{BX}$ или $Y_H \rightarrow I_0$ как проективному преобразованию, что показано стрелками. Определим выражение $Y_{BX}(Y_H)$ через параметры передачи. Для этого представим систему уравнений четырехполюсника через параметры передачи:

$$\begin{bmatrix} U_0 \\ I_0 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} U_1 \\ I_1 \end{bmatrix} = \frac{1}{Y_{10}} \begin{bmatrix} Y_{11} & 1 \\ \Delta_Y & Y_{00} \end{bmatrix} \begin{bmatrix} U_1 \\ I_1 \end{bmatrix},$$

где $\Delta_Y = Y_{00}Y_{11} - (Y_{10})^2$ – определитель матрицы Y -параметров.

Отсюда следуют известные дробно-линейные выражения:

$$Y_{BX} = \frac{I_0}{U_0} = \frac{a_{22}Y_H + a_{21}}{a_{12}Y_H + a_{11}}; \quad I_0 = U_0 \frac{a_{22}Y_H + a_{21}}{a_{12}Y_H + a_{11}}. \quad (7)$$

Аналогично (5) составим сложное отношение для входа и выхода четырехполюсника:

$$m_H^1 = (Y_H^{SC} \ Y_H^1 \ Y_H^{REF} \ Y_H^{OC}) = \frac{Y_H^1 - Y_H^{SC}}{Y_H^1 - Y_H^{OC}} \div \frac{Y_H^{REF} - Y_H^{SC}}{Y_H^{REF} - Y_H^{OC}} =$$

$$= n_H^1 \div n_H^{REF} = (I_0^{SC} \ I_0^1 \ I_0^{REF} \ I_0^{OC}) = m_0^1. \quad (8)$$

Как видно, сложное отношение m_H^1 является отношением простых отношений или обычной пропорции для текущей нагрузки n_H^1 и опорной нагрузки n_H^{REF} .

Полученные выражения явно не содержат параметры четырехполюсника. Поэтому можно вычислить проводимость нагрузки по измеренным токам на входе для четырехполюсника с изменяемыми параметрами. Рассмотрим цепь на рис. 4 как модель нестабильной двухпроводной линии питания силовой нагрузки или резистивного датчика.

Генератор управления ГЕН по очереди за достаточно короткое время подключает три базовых проводимости $Y_H^{SC}, Y_H^{REF}, Y_H^{OC}$ и саму нагрузку Y_H^1 . По из-

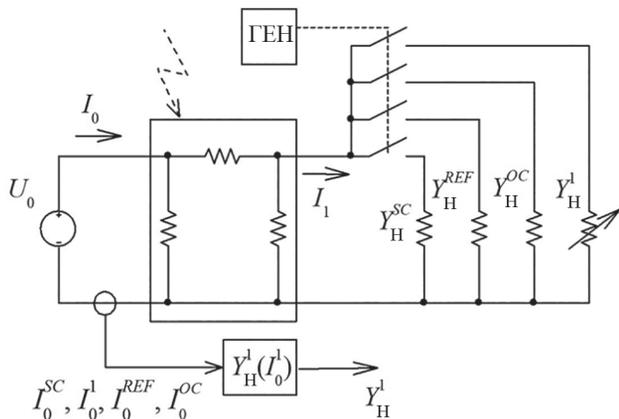


Рис. 4. Система измерения проводимости нагрузки Y_H^1

Fig. 4. Measurement system of load conductivity Y_H^1

меренным токам на входе $I_0^{SC}, I_0^1, I_0^{REF}, I_0^{OC}$ вычисляем $m_0^1 = m_H^1$ согласно (8). Для известных базовых проводимостей проводимость нагрузки

$$Y_H^1 = \frac{m_0^1 n_H^{REF} Y_H^{OC} - Y_H^{SC}}{m_0^1 n_H^{REF} - 1}. \quad (9)$$

Отсюда следует, что возможные отклонения заявленных значений базовых проводимостей приводят к погрешностям расчета проводимости нагрузки.

Возможен второй вариант измерения проводимости нагрузки. Принимаем величину сложного отношения

непосредственно за величину проводимости нагрузки $m_0^1 = Y_H^1$. Тогда к линии подключается предварительно подготовленное информационное или сигнальное значение этой нагрузки Y_S^1 . Поэтому сложное отношение как проводимость нагрузки имеет вид:

$$m_H^1 = Y_H^1 = (Y_H^{SC} Y_S^1 Y_H^{REF} Y_H^{OC}) = n_S^1 \div n_H^{REF} = (I_0^{SC} I_0^1 I_0^{REF} I_0^{OC}).$$

Само значение Y_S^1 определяется из этого сложного отношения аналогично (9).

Цель с двумя нагрузками. Проективные координаты рабочей точки на плоскости. Рассмотрим активный четырехполюсник A на рис. 5. Для указанных направлений токов уравнение этой цепи имеет вид:

$$\begin{bmatrix} I_1 \\ I_2 \end{bmatrix} = \begin{bmatrix} -Y_{11} & Y_{12} \\ Y_{12} & -Y_{22} \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} + \begin{bmatrix} I_1^{SC} \\ I_2^{SC} \end{bmatrix}.$$

С учетом напряжения нагрузок $U_1 = I_1/Y_{H1}, U_2 = I_2/Y_{H2}$ получаем выражения двух пучков нагрузочных характеристик $(I_1, I_2, Y_{H1}) = 0, (I_1, I_2, Y_{H2}) = 0$ как прямых линий. Проводимости Y_{H1}, Y_{H2} являются параметрами пучков. Центры этих пучков определяют дополнительные характерные режимы. В частности, центр пучка G_1 не зависит от проводимости Y_{H2} , что соответствует току $I_2 = 0$ и параметрам первой нагрузки $Y_{H1} = Y_{H1}^{G1} = I_1^{G1} / U_1^{G1} < 0$. Аналогично центр пучка G_2 определяется параметрами второй нагрузки $Y_{H2} = Y_{H2}^{G2} = I_2^{G2} / U_2^{G2} < 0$.

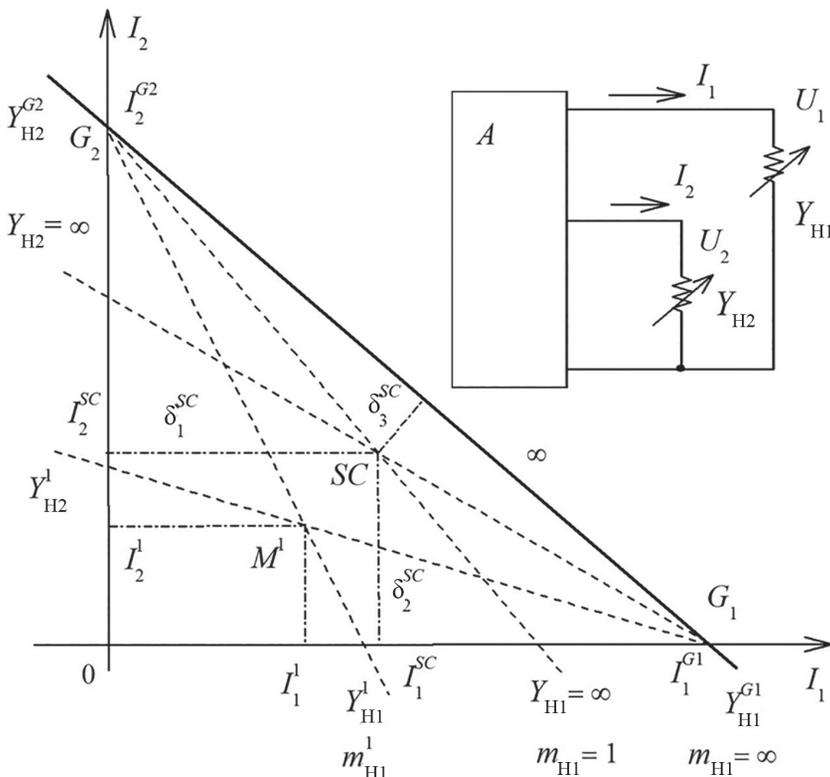


Рис. 5. Активный четырехполюсник и два пучка нагрузочных линий с параметрами Y_{H1}, Y_{H2}

Fig. 5. Active two-port and two bunches of load straight lines with parameters Y_{H1}, Y_{H2}

Полученный координатный треугольник $G_1 0 G_2$ и точка SC короткого замыкания как единичная точка задают проективную систему координат. Прямая $G_1 G_2$ является бесконечно удаленной прямой линией. Тогда проявляется по три характерных или базовых значения нагрузок: $Y_{H1} = 0, Y_{H1} = \infty, Y_{H1}^{G1}$; $Y_{H2} = 0, Y_{H2} = \infty, Y_{H2}^{G2}$. Обратим внимание, что в отличие от цепи с одной нагрузкой такие базовые значения, как $Y_{H1} = 0, Y_{H1}^{G1}$ и $Y_{H2} = 0, Y_{H2}^{G2}$, задаются однозначно, так как определяют стороны координатного треугольника. Только единичную точку можно взять произвольным или удобным образом, необязательно как режим короткого замыкания.

Выразим неоднородные координаты для текущих значений нагрузок как сложное отношение:

$$m_{H1}^1 = (0 Y_{H1}^1 \infty Y_{H1}^{G1}) = \frac{Y_{H1}^1}{Y_{H1}^1 - Y_{H1}^{G1}} = \frac{\xi_{H1}^1}{\xi_{H3}^1};$$

$$m_{H2}^1 = (0 Y_{H2}^1 \infty Y_{H2}^{G2}) = \frac{Y_{H2}^1}{Y_{H2}^1 - Y_{H2}^{G2}} = \frac{\xi_{H2}^1}{\xi_{H3}^1}. \quad (10)$$

В свою очередь, однородные координаты

$$\xi_{H1}^1 = \frac{\delta_1^1}{\delta_{SC}^1} = \frac{I_1^1}{I_{SC}^1}; \quad \xi_{H2}^1 = \frac{\delta_2^1}{\delta_{SC}^1} = \frac{I_2^1}{I_{SC}^1}; \quad \xi_{H3}^1 = \frac{\delta_3^1}{\delta_{SC}^1} \quad (11)$$

определяются отношением расстояний точки текущего режима M^1 и точки SC до сторон координатного треугольника.

Инварианты вход–выход шестиполюсника. Рассмотрим отображение совмещенных нагрузочных характе-

ристик на выходе $(I_1, I_2, Y_{H1}) = 0, (I_1, I_2, Y_{H2}) = 0$ и входе $(I_3, I_4, Y_{H1}) = 0, (I_3, I_4, Y_{H2}) = 0$ шестиполюсника на рис. 6.

Такое отображение соответствует проективному преобразованию на плоскости. Поэтому, координатный треугольник $G_1 0 G_2$, точка текущего режима M^1 и точка SC соответствуют треугольнику $\bar{G}_1 \bar{0} \bar{G}_2$, точке \bar{M}^1 и точке \bar{SC} . Тогда оси токов I_1, I_2 соответствуют осям \bar{I}_1, \bar{I}_2 .

Точка \bar{M}^1 определяется фактическими токами I_1^1, I_2^1 . С другой стороны, эта точка определяется проективными неоднородными m_{BX1}^1, m_{BX2}^1 и однородными координатами $\bar{\xi}_1^1, \bar{\xi}_2^1, \bar{\xi}_3^1$ относительно координатного треугольника $\bar{G}_1 \bar{0} \bar{G}_2$, точки \bar{SC} и бесконечно удаленной прямой $\bar{G}_1 \bar{G}_2$. Для проективных преобразований проективные координаты точки M^1 равны координатам точки \bar{M}^1 , т.е. $m_{BX1}^1 = m_{H1}^1, m_{BX2}^1 = m_{H2}^1$.

Если провести все расчеты, то неоднородные координаты для текущих значений нагрузок выражаются через токи на входе соответствующими дробно-линейными соотношениями аналогично (10, 11):

$$m_{H1}^1 = m_{BX1}^1 = \frac{\bar{\xi}_1^1}{\bar{\xi}_3^1}; \quad m_{H2}^1 = m_{BX2}^1 = \frac{\bar{\xi}_2^1}{\bar{\xi}_3^1}. \quad (12)$$

Однородные координаты

$$\bar{\xi}_1^1 = \frac{\bar{\delta}_1^1}{\bar{\delta}_{SC}^1}; \quad \bar{\xi}_2^1 = \frac{\bar{\delta}_2^1}{\bar{\delta}_{SC}^1}; \quad \bar{\xi}_3^1 = \frac{\bar{\delta}_3^1}{\bar{\delta}_{SC}^1} \quad (13)$$

определяются отношением расстояний точки текущего режима \bar{M}^1 и точки \bar{SC} до сторон координатного тре-

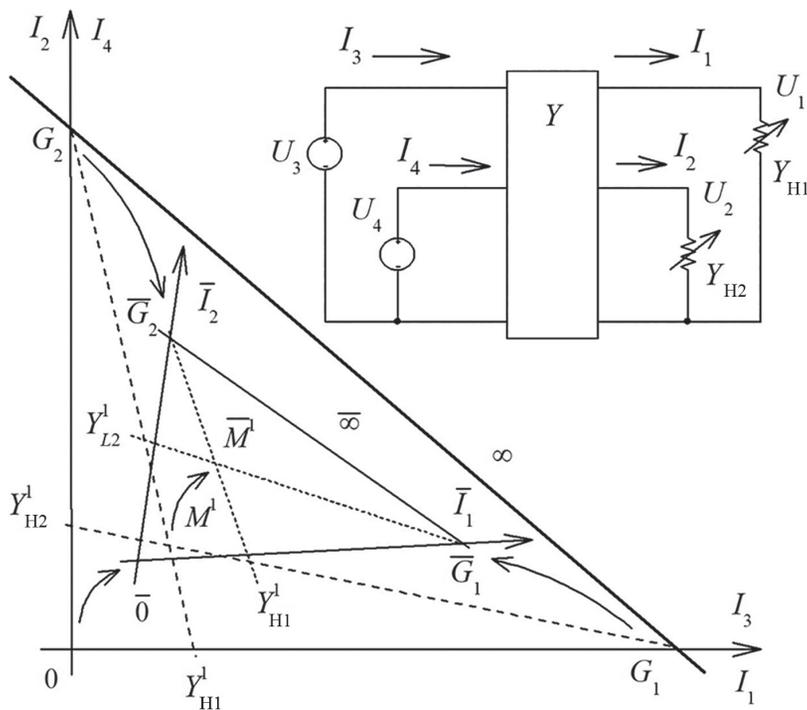


Рис. 6. Отображение нагрузочных характеристик в плоскости (I_1, I_2) на плоскость (I_3, I_4)

Fig. 6. Mapping of load characteristics on the plane (I_1, I_2) to the plane (I_3, I_4)

угольника $\bar{G}_1 \bar{0} \bar{G}_2$. В отличие от цепи с одной нагрузкой координаты (10–13) явно содержат параметры шести-полюсника. В случае изменения его параметров необходимо постоянно переопределять параметры координатных треугольников и однородные координаты.

Рассмотрим идею реализации измерения на рис. 7.

Подобная цепь может быть моделью нестабильной трехпроводной линии питания как силовых нагрузок, так и резистивных датчиков. Принцип работы аналогичен цепи с одной нагрузкой. Отличие заключается в необходимости этапа предварительного тестирования выхода линии для определения базовых значений Y_{H1}^{G1} , Y_{H2}^{G2} с использованием еще дополнительно введенных величин Y_{H1}^{REF} , Y_{H2}^{REF} . Для этого генератор управления подключает базовые проводимости согласно комбинациям табл. 1.

Рассмотрим точку тестового режима REF . Вводим однородные проективные координаты ξ_1^{REF} , ξ_2^{REF} , ξ_3^{REF} точки REF через отношение её расстояний δ_1^{REF} , δ_2^{REF} , δ_3^{REF} и расстояний δ_1^{SC} , δ_2^{SC} , δ_3^{SC} точки до сторон координатного треугольника $G_1 0 G_2$, используя данные табл. 1:

$$\xi_1^{REF} = \frac{\delta_1^{REF}}{\delta_1^{SC}} = \frac{I_1^{REF}}{I_1^{SC}}; \xi_2^{REF} = \frac{\delta_2^{REF}}{\delta_2^{SC}} = \frac{I_2^{REF}}{I_2^{SC}}; \xi_3^{REF} = \frac{\delta_3^{REF}}{\delta_3^{SC}}. \tag{14}$$

Неоднородные проективные координаты m_1^{REF} , m_2^{REF} точки REF :

$$m_1^{REF} = \frac{\xi_1^{REF}}{\xi_3^{REF}}; m_2^{REF} = \frac{\xi_2^{REF}}{\xi_3^{REF}}. \tag{15}$$

С другой стороны, неоднородные проективные координаты задаются сложным отношением (5):

$$m_1^{REF} = (Y_{H1}^{OC} Y_{H1}^{REF} Y_{H1}^{SC} Y_{H1}^{G1}) = \frac{Y_{H1}^{REF}}{Y_{H1}^{REF} - Y_{H1}^{G1}}; m_2^{REF} = \frac{Y_{H2}^{REF}}{Y_{H2}^{REF} - Y_{H2}^{G2}}. \tag{16}$$

Отсюда находим значения проводимостей:

$$Y_{L1}^{G1} = Y_{L1}^{REF} \frac{m_1^{REF} - 1}{m_1^{REF}}; Y_{L2}^{G2} = Y_{L2}^{REF} \frac{m_2^{REF} - 1}{m_2^{REF}}. \tag{17}$$

Блок расчета опорных проводимостей работает по формулам (14–17).

Рассмотрим токи на входе линии. Обозначения токов на входе и соответствие токам на выходе представлены в табл. 2.

Определяем вершины и стороны координатного треугольника, используя значения входных токов и рис. 6. Этап предварительного тестирования и расчетов на этом заканчивается.

Аналогично цепи с одной нагрузкой применим второй вариант измерения проводимости нагрузки. Принимаем сложные отношения (10) непосредственно за проводимости нагрузок $m_{H1}^1 = Y_{H1}^1$, $m_{H2}^1 = Y_{H2}^1$. Тогда сигнальные проводимости рассчитываются по этим передаваемым сигналам:

$$Y_{S1}^1 = \frac{Y_{H1}^1 Y_{H1}^{G1}}{Y_{H1}^1 - 1}; Y_{S2}^1 = \frac{Y_{H2}^1 Y_{H2}^{G2}}{Y_{H2}^1 - 1}.$$

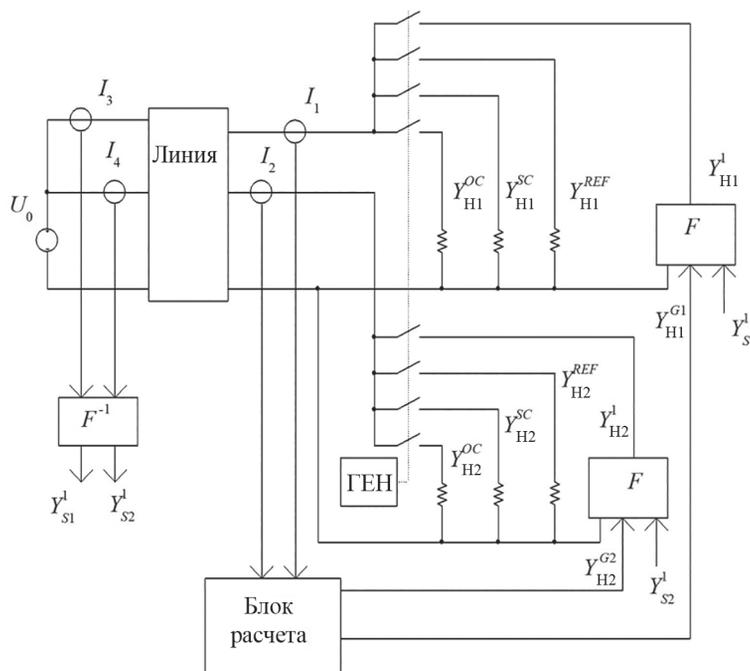


Рис. 7. Система измерения проводимости датчиков Y_{S1}^1, Y_{S2}^1

Fig. 7. Measurement system of sensor conductivities Y_{S1}^1, Y_{S2}^1

Таблица 1

Соответствие комбинаций базовых проводимостей и выходных токов

Correspondence between load conductivities and output currents

Наборы проводимостей		Токи на выходе	
$Y_{L1}^{OC} = 0$	$Y_{L2}^{OC} = 0$	$I_1^{OC} = 0$	$I_2^{OC} = 0$
$Y_{L1}^{OC} = 0$	$Y_{L2}^{SC} = \infty$	$I_1^{OC,SC} = 0$	$I_2^{OC,SC}$
$Y_{L1}^{SC} = \infty$	$Y_{L2}^{OC} = 0$	$I_1^{SC,OC}$	$I_2^{SC,OC} = 0$
$Y_{L1}^{SC} = \infty$	$Y_{L2}^{SC} = \infty$	I_1^{SC}	I_2^{SC}
Y_{L1}^{REF}	Y_{L2}^{REF}	I_1^{REF}	I_2^{REF}

Таблица 2

Соответствие комбинаций токов на выходе и входе

Correspondence between output and input currents

Токи на выходе		Токи на входе	
$I_1^{OC} = 0$	$I_2^{OC} = 0$	I_3^{OC}	I_4^{OC}
$I_1^{OC,SC} = 0$	$I_2^{OC,SC}$	$I_3^{OC,SC}$	$I_4^{OC,SC}$
$I_1^{SC,OC}$	$I_2^{SC,OC} = 0$	$I_3^{SC,OC}$	$I_4^{SC,OC}$
I_1^{SC}	I_2^{SC}	I_3^{SC}	I_4^{SC}
I_1	I_2	I_3	I_4

Блоки F работают по этим формулам. На входе линии, используя данные табл. 2, блок F^{-1} рассчитывает однородные проективные координаты $\bar{z}_1^1, \bar{z}_2^1, \bar{z}_3^1$ и неоднородные координаты (12, 13) как непосредственно проводимости нагрузок.

Расчет сопротивления нагрузки четырехполюсника нейронной сетью. Рассмотренные примеры применения инвариантных свойств вход–выход многополюсников (даже с двумя нагрузками) приводят к громоздким выражениям. Пример подобных выражений для трех нагрузок приведен в [4]. Кроме того, необходимо переопределять изменяемые параметры этих выражений. Все это усложняет и ограничивает возможности практической реализации. Использование аппарата искусственных нейронных сетей позволяет принципиально снять эти проблемы. Рассмотренные примеры относятся к задачам аппроксимации или регрессии. В работе используется пакет *Deep Learning Toolbox* системы *MATLAB*, версия *R2021b*. Можно рекомендовать основы нейронных сетей на языке Python с библиотекой *Keras* [15]. В этой книге приведен близкий пример регрессии.

Задачи аппроксимации решаются поверхностной (*Shallow*) нейронной сетью. Для начала удобнее использовать графический интерфейс *GUI* пользователя. Для этого вводится команда `nstart` в командной строке запущенной программы *MATLAB*. В раскрытом окне *GUI* выбирается опция *Fitting app*. В следующем окне предлагается структура нейронной сети прямого распространения с одним скрытым слоем (*Hidden Layer*) с нелинейной сигмоидной функцией активации и вы-

ходным слоем с линейной функцией активации. Точность аппроксимации определяется числом нейронов скрытого слоя и соответствующими размерами вектора обучения и вектора цели, которые определяют базу данных обучения.

Подготовка базы данных четырехполюсника. Рассмотрим пример четырехполюсника на рис. 3. Примем значения безразмерных параметров: $y_0 = 1, y_1 = 2, 4$.

Пусть значения проводимости y_{10} изменяются в достаточно большом диапазоне 4–7 с шагом 0,2, что дает 16 выборок. Для удобства здесь и далее будем использовать представление переменных в системе *MATLAB*. Тогда $y_{10} = 4:0,2:7$. Диапазон изменения значений базовых проводимостей примем более узким, но также с 16-ю выборками:

$$Y_H^{OC} = 2,025 : 0,025 : 2,4; Y_H^{REF} = 10,125 : 0,125 : 12;$$

$$Y_H^{SC} = 20,25 : 0,25 : 24.$$

В свою очередь, проводимость непосредственно измерительной нагрузки Y_H^{MES} изменяется в широкой области, причем эта область разбита на два диапазона по восемь выборок. В одном из них для $Y_H^{OC} < Y_H^{MES} < Y_H^{REF}$ положим $Y_H^{MES} = 2,5:1:9,5$. В другом диапазоне для $Y_H^{REF} < Y_H^{MES} < Y_H^{SC}$ примем $Y_H^{MES} = 12,5:1:19,5$.

Входная проводимость четырехполюсника определяется выражением

$$Y_{BX} = \frac{3,4y_{10} + (1 + y_{10})Y_H}{y_{10} + Y_H + 2,4}. \quad (18)$$

В свою очередь, для $U_0 = 1$ входной ток $I_0 = Y_{BX}$.

Теперь необходимо рассчитать вектор обучения или вектор входного тока (вектор-столбец) с компонентами:

$$I_0 = [I_0^{OC}; I_0^{REF}; I_0^{SC}; I_0^{MES}]. \quad (19)$$

Рассчитаем все выборки для компоненты I_0^{OC} . Для этого в выражении (18) для Y_{BX} подставляем по очереди все выборки Y_H^{OC} для очередной выборки y_{10} . Для проведения расчетов удобно организовать *Script*-файл с помощью соответствующего редактора. Программа расчета использует цикл *for* для вектора Y_H^{OC} , вложенный цикл *for* для y_{10} и пользовательскую функцию (18). Таким образом, взаимный перебор значений Y_H^{OC}, y_{10} дает $16 \times 16 = 256$ комбинаций тока I_0^{OC} , которые выводятся из этих циклов в виде матрицы или вектора. Аналогично получаем по 256 комбинаций для остальных базовых компонент I_0^{REF}, I_0^{SC} .

В отношении расчета измерительной компоненты тока I_0^{MES} и вектора цели tt : через такой же вложенный цикл *for* для y_{10} повторяем все выборки измерительной нагрузки Y_H^{MES} , чтобы было соответствие получаемого вектора цели tt с выборками измерительного тока I_0^{MES} . Эти векторы по 256 элементов выводятся из этого цикла.

В *Script*-файле в конце текста программы вводится сама пользовательская функция (18). В конечном итоге

вектор-столбец входного тока I_0 согласно (19) содержит четыре строки по 256 элементов или выборок. Поэтому размер выборки входного вектора равен четырем. Необходимо предварительно нормировать все элементы, например по максимальному значению.

Для проверки нейронной сети в процессе обучения подготовим контрольную базу данных. Примем по четыре выборки всех проводимостей, значения которых должны находиться в диапазоне изменения проводимостей для обучения, но не совпадать с ними. Пусть

$$\begin{aligned} Y_{10}^{TST} &= [4, 1; 5, 1; 5, 9; 6, 9]; \\ Y_{TST}^{OC} &= [2, 035; 2, 16; 2, 26; 2, 385]; \\ Y_{TST}^{REF} &= [10, 18; 10, 56; 11, 31; 11, 92]; \\ Y_{TST}^{SC} &= [20, 35; 21, 6; 22, 6; 23, 85]; \\ Y_H^{TST} &= [4, 0; 8, 0; 13, 0; 18, 0]. \end{aligned}$$

Итак, контрольный вектор-столбец входного тока I_0^{TST} содержит также четыре строки по $4 \times 4 = 16$ элементов. Контрольный вектор цели также содержит 16 элементов. Для проведения расчетов надо вводить переменные в формате *MATLAB* без верхних и нижних индексов. Например, I_0^{TST} заменяем на $I0ts_16$, Y_H^{TST} на Yts и т.д. После выполнения расчетов базы обучения и контрольной базы данные находятся в рабочей области *Workspace*. Далее эти векторы сохраняются в организованной папке.

Вернемся к *GUI*. Следующий шаг связан с загрузкой вектора обучения и вектора цели из рабочей области. Далее данные предлагается разделить на обучающие, проверочные и тестовые с указанными процентами, чтобы исключить переобучение [8]. На следующем шаге необходимо выбрать число нейронов скрытого слоя. Точных формул расчета этого числа нет, а практический опыт подсказывает, что для выборки входного вектора, равной четырем, число нейронов скрытого слоя может быть от четырех до восьми. Далее переходим непосредственно к обучению с возмож-

ностью выбора алгоритма обучения. Результаты обучения представляются в отдельном окне. Раскрывая предлагаемые графики, можно оценить точность аппроксимации обученной сети. Следующие шаги позволяют сохранить *Script*-файл программы обучения сети с подходящим именем и *Simulink*-файл модели сети. На этом можно закрыть *GUI*.

Результаты обучения и моделирования нейронной сети. Чтобы провести численные эксперименты по обучению и проверить точность сети, открываем сохраненные *Script*-файл, файлы данных обучения и контроля. В программу, представленную *Script*-файлом, вводим дополнительные команды, например моделирование сетью *net* контрольного вектора входного тока:

$$Ysim = sim(net, I0ts_16).$$

Результаты моделирования *Ysim* представляем относительной погрешностью

$$e_Ysim = [ts; Ysim; 100 * (ts - Ysim) ./ ts].$$

Примем число нейронов равным шести, т.е. $hiddenLayerSize = 6$. График хода обучения показывается в специальном окне. Процесс обучения заканчивается при выполнении числа заданных эпох при достижении заданной среднеквадратичной ошибки *mse* или при увеличении погрешности проверочных данных относительно обучаемых данных. После нескольких повторений обучения получен приемлемый результат. Изменение среднеквадратичной ошибки в процессе обучения показано на рис. 8.

Окончательная оценка проводится по фактическим данным относительной погрешности e_Ysim , представленным в формате *MATLAB*:

$e_Ysim =$			
Columns 1 through 4			
4.0000	4.0000	4.0000	4.0000
3.9880	3.9906	3.9919	3.9948
0.2995	0.2353	0.2036	0.1288

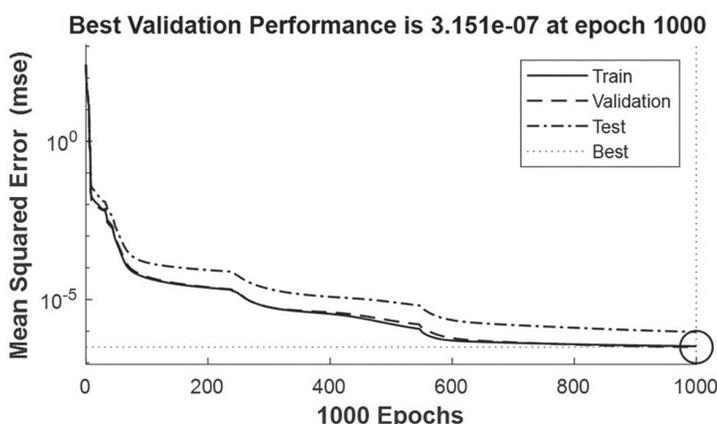


Рис. 8. Изменение среднеквадратичной ошибки в процессе обучения

Fig. 8. Change the mean-squared error in the training process

Columns 5 through 8

8.0000	8.0000	8.0000	8.0000
8.0623	8.0286	8.0027	7.9739
-0.7787	-0.3579	-0.0337	0.3259

Columns 9 through 12

13.0000	13.0000	13.0000	13.0000
12.9803	12.9847	12.9872	12.9895
0.1518	0.1177	0.0984	0.0811

Columns 13 through 16

18.0000	18.0000	18.0000	18.0000
17.9671	17.9639	17.9630	17.9621
0.1829	0.2005	0.2058	0.2108

Первые строки определяют контрольный вектор цели I_0 или фактическую нагрузку. Вторые строки соответствуют расчетной нагрузке. Третьи строки являются относительной погрешностью в процентах. Как видно, можно получить погрешность менее одного процента. Примерно такие же результаты после многократных попыток обучения получаются для восьми и даже четырех нейронов. Далее генерируется или выводится модель полученной сети в *Simulink*-редакторе с помощью команды *gensim(net)* в командной строке.

Расчет сопротивления двух нагрузок многополюсника нейронной сетью. Подготовка базы данных. Рассмотрим пример цепи на рис. 9. Для такой цепи удобнее перейти от проводимостей к сопротивлению нагрузок. Из-за сопротивления общего провода r_{34} наблюдается взаимовлияние токов на входе и выходе цепи при изменении сопротивления нагрузок. Примем, что сопротивление r_{34} также может меняться. Как было сказано выше, необходимо по известным токам на входе определить сопротивления нагрузок.

Примем безразмерные значения неизменяемых параметров:

$$U_3 = 12; U_4 = 10; r_3 = 3; r_1 = 1; r_{31} = 9; r_4 = 4; r_2 = 2; r_{42} = 12.$$

Пусть диапазон изменения сопротивления $r_{34} = 1,025:0,05:1,375$, что соответствует более 25 % и определяет восемь выборок. Диапазон изменения значений базовых сопротивлений примем более узким, но также с восемью выборками согласно табл. 3. В свою очередь, сопротивления непосредственно измерительных нагрузок $R_{H1}^{MES}, R_{H2}^{MES}$ изменяются в широкой области, но не включают в себя значения соответствующих опорных сопротивлений $R_{H1}^{REF}, R_{H2}^{REF}$.

Теперь с учетом табл. 1 и 2 необходимо рассчитать вектор обучения с 12 компонентами:

$$I_0 = [I_3^{OCOC}; I_3^{OCSC}; I_3^{REF}; I_3^{SCSC}; I_3^{SCOC}; I_4^{OCOC}; I_4^{OCSC}; I_4^{REF}; I_4^{SCSC}; I_4^{SCOC}; I_3^{MES}; I_4^{MES}]. \quad (20)$$

Для этого используем выражения для входных токов, полученных из расчета нашей цепи методом контурных токов [16]. Организовываем *Script*-файл и в формате *MATLAB* объявляем символьные переменные и составляем программу расчета:

```
syms I1 I2 I3 I4 r34 RH1 RH2;
eq1 = I3*(3 + 9 + r34) - I1*9 + I4*r34 - 12;
eq2 = I4*(4 + 12 + r34) - I2*12 + I3*r34 - 10;
eq3 = -I3*9 + I1*(1 + RH1 + 9);
eq4 = -I4*12 + I2*(1 + RH2 + 12);
eqns = [eq1, eq2, eq3, eq4];
s = solve(eqns, I1, I2, I3, I4); I3 = s.I3; I4 = s.I4.
```

Приведем полученные выражения для токов с операторами поэлементного умножения и деления:

$$I3 = 2*(RH1 + 10).*(96*RH2 + I4*r34 + RH2.*r34 + 480)./det; \quad (21)$$

$$I4 = 2*(RH2 + 14).*(60*RH1 - 10*r34 - RH1.*r34 + 195)./det; \quad (22)$$

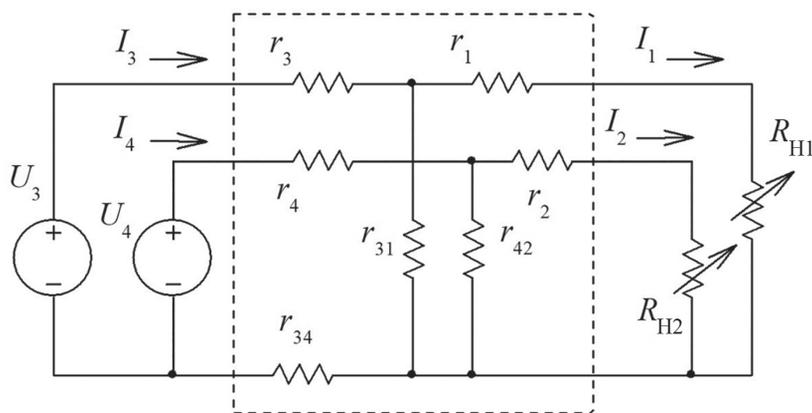


Рис. 9. Многополюсник с двумя нагрузками и с сопротивлением r_{34} общего провода

Fig. 9. Multiport with two loads and resistance r_{34} of the common wire

Таблица 3

**Изменения значений базовых сопротивлений
и нагрузок**

Changes of base resistances and loads

Нагрузка	Значение		Изменение	
	мин	макс	шаг	%
R_{H1}^{OC}	20,25	23,75	0,50	14,7
R_{H2}^{OC}	30,25	33,75	0,50	10,3
R_{H1}^{REF}	8,025	8,375	0,05	4,20
R_{H2}^{REF}	15,05	15,75	0,10	4,40
R_{H1}^{SC}	4,025	4,375	0,05	8,00
R_{H2}^{SC}	6,025	6,375	0,05	5,40
R_{H1}^{MES}	5,000	16,20	1,60	69,1
R_{H2}^{MES}	7,000	26,60	2,80	73,7

$$\begin{aligned} \det = & 960 * RH1 + 624 * RH2 + \\ & + 1346 * r34 + 192 * RH1 * RH2 + \dots \\ & 248 * RH1 * r34 + 199 * RH2 * r34 + \\ & + 28 * RH1 * RH2 * r34 + 3120. \end{aligned}$$

Рассчитаем все выборки для парных компонент I_3^{OC} и I_4^{OC} . Программа расчета использует цикл *for* для вектора R_{H1}^{OC} , вложенный цикл *for* для R_{H2}^{OC} и следующий вложенный цикл *for* для $r34$ и пользовательские функции (21, 22). Таким образом, взаимный перебор значений R_{H1}^{OC} , R_{H2}^{OC} и $r34$ дает $8 \times 8 \times 8 = 512$ комбинаций для каждого тока, которые выводятся из этих циклов. Аналогично получаем по 512 комбинаций для остальных парных базовых компонент из (20).

В отношении векторов измерительных компонент токов I_3^{MES} , I_4^{MES} и вектора цели $tt = [tt1; tt2]$. Вектор цели состоит из двух векторов $tt1$, $tt2$ для каждой нагрузки. Также используем цикл *for* для вектора R_{H1}^{MES} , вложенный цикл *for* для R_{H2}^{MES} и следующий вложенный цикл *for* для $r34$ и пользовательские функции (21, 22), чтобы было соответствие с выборками базовых компонент и измерительных токов. В конечном итоге вектор-столбец входного тока I_0 (заменяем на I_0_{512}) согласно (20) содержит 12 строк по 512 выборок. Поэтому размер выборки входного вектора равен 12.

Для проверки нейронной сети в процессе обучения подготовим тестовую базу данных. Примем по три выборки всех сопротивлений:

$$\begin{aligned} r_{34}^{TST} &= [1,035; 1,175; 1,315]; \\ R_{TST1}^{OC} &= [20,35; 21,75; 23,15]; \\ R_{TST2}^{OC} &= [30,35; 31,75; 33,15]; \\ R_{TST1}^{REF} &= [8,035; 8,175; 8,315]; \\ R_{TST2}^{REF} &= [15,085; 15,365; 15,645]; \\ R_{TST1}^{SC} &= [4,035; 4,175; 4,315]; \\ R_{TST2}^{SC} &= [6,035; 6,175; 6,315]; \\ R_{TST1}^{MES} &= [5,5; 9,9; 14,3]; R_{TST2}^{MES} = [7,5; 15,5; 23,5]. \end{aligned}$$

В итоге, повторив расчеты по той же программе, получим тестовый вектор-столбец входного тока I_0^{TST} (заменяем на I_0ts_{27}), который содержит также 12 строк по $3 \times 3 \times 3 = 27$ элементов. Тестовый вектор цели $ts = [ts1; ts2]$ содержит две строки.

Результаты обучения и моделирования нейронной сети. В программу, представленную *Script*-файлом, вводим необходимые данные и команды. В данном случае можно обучать нейронную сеть сразу для двух нагрузок или по отдельности для каждой нагрузки, но для двух независимых сетей. В любом случае вводим вектор обучения I_0_{512} и контрольный вектор I_0ts_{27} . В случае двух нагрузок вводим вектор цели tt и контрольный вектор цели ts из двух строк. Тогда сеть будет с двумя выходами. Если сеть обучается по каждой нагрузке отдельно, то вводим вектор цели $tt1$ или $tt2$. Аналогично вводим контрольный вектор цели $ts1$ или $ts2$. Результаты обучения сети с двумя нагрузками также представляем относительной погрешностью по каждой нагрузке. В случае сети с одной нагрузкой относительная погрешность представляется по соответствующей нагрузке.

Обучаем сеть с двумя нагрузками. Примем число нейронов равным 15. Изменение среднеквадратичной ошибки в процессе обучения аналогично рис. 8. А относительная ошибка по каждой из нагрузок не превышает 2,5 %. Численные эксперименты для 10–20 нейронов показывают примерно такие же результаты. Обучение сети отдельно по каждой нагрузке для 6–8 нейронов приводит к относительной ошибке не более 1,5 %.

Обсуждение полученных результатов. Представленные численные эксперименты для сетей с одной и двумя нагрузками подтверждают совокупный эффект применения нейронных сетей с инвариантными свойствами многополюсников в задачах расчета или измерения нагрузок. Поэтому вектор обучения содержит необходимое число компонент. Для сравнения, в случае трех нагрузок вектор обучения уже состоит из 32 компонент. Чтобы подтвердить, что нейронная сеть неявно реализует инвариантные свойства четырехполюсника, были проведены численные эксперименты по обучению сети вектором обучения, содержащим только измерительную компоненту I_0^{MES} . Ошибки контрольного моделирования составили десятки и более процентов. Аналогичные результаты получены и для двух нагрузок.

Рассмотрим особенности результатов обучения сети. Обращает внимание такая выявленная закономерность, что значения относительной погрешности e_{Ysim} могут изменяться в больших пределах (наглядно видно для столбцов 5 и 7 в случае одной нагрузки). Это может говорить о проявлении эффекта переобучения. Эффективный способ улучшения качества обучения связан с прерыванием процесса обучения на каком-то числе эпох. Интерфейс *GUI* позволяет это

выполнить. Для этого наблюдается график изменения среднеквадратичной ошибки, аналогичный рис. 8. Но этот открытый график замедляет выполнение программы обучения. Также известен ряд специальных методов по подавлению переобучения, но это выходит за рамки изложенного начального опыта применения нейронной сети.

Для сети с двумя нагрузками значения относительной погрешности могут отличаться между нагрузками. Чтобы получить приемлемый результат, приходится многократно повторять обучение. Если сеть обучается отдельно по каждой нагрузке, то ошибки заметно меньше. В отношении увеличения размера вектора обучения: если взять по 16 выборок, то получается $16 \times 16 \times 16 = 4096$ комбинаций. Обучение в этом случае не дает заметного эффекта, как и увеличение числа нейронов скрытого слоя более 20. По умолчанию в *Script*-файле программы обучения используется один скрытый слой. Введение двух скрытых слоёв с числом нейронов от 5 до 10 не дает заметного эффекта.

Для составления базы данных можно использовать непосредственно результаты моделирования цепи в *Simulink*-редакторе или, используя опцию *Import*, импортировать данные, например из программы *Multisim* [17]. Но с учетом взаимного изменения трех (возможно и более) величин такой несложной цепи с двумя нагрузками удобнее использовать представленные аналитические выражения, полученные в результате символического расчета.

Выводы. Известные подходы к расчету сопротивления нагрузок многополюсника с изменяемыми параметрами приводят к сложным выражениям и необходимости переопределения параметров в этих выражениях в рабочем режиме. Все это ведет к усложнению возможной аппаратной реализации расчетного блока и затратам времени расчета, особенно при увеличении числа нагрузок.

Применение обученной нейронной сети позволяет принципиальным образом упростить расчет и обеспечивает необходимую точность.

Подготовка данных обучения, тестирования и сам процесс обучения проводятся предварительно и не связаны с рабочим режимом работы.

Полученные результаты дают основу для рассмотрения как практически более важных цепей, так и алгоритмов обучения нейронных сетей.

Возможно применение соответствующих нейронных сетей для цепей переменного тока ввиду выпол-

нения инвариантов вход–выход многополюсников для комплексных величин.

СПИСОК ЛИТЕРАТУРЫ

1. Бессонов Л.А. Теоретические основы электротехники. Электрические цепи. М.: Высшая школа, 1996, 640 с.
2. Bhattacharyya S.P., Keel L.H., Mohsenizadeh D.N. Linear Systems: A Measurement-Based Approach. Springer, India, 2014, 89 p.
3. Пенин А.А. Проективная геометрия в теории электрических цепей с переменными и нелинейными нагрузками: научный реф. дис. ... доктора техн. наук. Кишинев, 2017, 81 с.
4. Penin A. Analysis of Electrical Circuits with Variable Load Regime Parameters: Projective Geometry Method. Springer International Publishing, Cham, Switzerland, 2020, 520 p., DOI:10.1007/978-3-030-35366-7.
5. Мазин В.Д. Способ повышения точности измерительных приборов и преобразователей. – Измерительная техника, 1980, № 6, с. 14–15.
6. Mazin V.D. Error of Measurement in the Compound-Ratio Method. – Measurement Techniques, 1983, vol. 26, No. 8, pp. 628–629, DOI: 10.1007/BF00833215.
7. Цыбульский О.А. Проективные свойства широкодиапазонных измерений. – Измерительная техника, 2013, № 1, с. 27–29.
8. Дьяконов В.П., Круглов В.В. MATLAB 6.5 SP1/7/7 SP1/7 SP2+ Simulink 5/6. Инструменты искусственного интеллекта и биоинформатики. М.: СОЛОН-ПРЕСС, 2009, 456 с.
9. Бураков М.В. Нейронные сети и нейроконтроллеры. СПб.: ГУАП, 2013, 284 с.
10. Вакуленко С.А., Жихарева А.А. Практический курс по нейронным сетям. СПб.: Университет ИТМО, 2018, 71 с.
11. Deep Learning Toolbox [Электрон. ресурс], URL: <https://www.mathworks.com/help/deeplearning/index.html> (Дата обращения 20.02.2022).
12. Глаголев Н.А. Проективная геометрия. М.: Высшая школа, 1963, 343 с.
13. Ayres F. Schaum's Outline Series Theory and Problems of Projective Geometry. New York, McGraw-Hill, 1967, 249 p.
14. Riaz R. Circuit theory in Projective Space and Homogeneous Circuit Models. – IEEE Transactions on Circuits and Systems I: Regular Papers, 2019, vol. 66, No. 2, pp. 463–476.
15. Шолле Ф. Глубокое обучение на Python. СПб.: Питер, 2018, 400 с.
16. Новгородцев А.Б. Расчет электрических цепей в MATLAB. СПб.: Питер, 2004, 250 с.
17. Jin Z., Kaba, S. Deep Neural Network Based Behavioral Model of Nonlinear Circuits. – Journal of Applied Mathematics and Physics, 2021, No. 9, pp. 403–412.

[22.02.2022]



Автор: Пенин Александр Анатольевич – доктор техн. наук, главный научный сотрудник Института электронной инженерии и нанотехнологий им. Д. Гишу, Кишинев, Республика Молдова.

Neural Network Based Calculation of Load Resistances Taking into Account the Multiport Input-to-Output Ratio Invariant Properties

PENIN Alexandr A. (*Ghitu Institute of the Electronic Engineering and Nanotechnologies, Kishinev, Moldova Republic*) – Chief Scientific, Dr. Sci. (Eng.).

The invariant properties of a multiport input-output ratio as a model of unstable two- and three-wire power supply lines of power loads or resistive sensors are considered. The fractional linear expressions typical for the electrical circuit theory are interpreted as projective transformations in the sense of projective geometry. Projective transformations preserve an invariant: a cross ratio or a duplicate proportion of four samples of variable resistance and the corresponding values of currents in different sections of the circuit. In this way, it becomes possible to calculate the load resistances from the measured current samples at the circuit input and their known base or test values; in doing so, the circuit parameters are not explicitly used. In turn, the base load values can also vary or are known with some uncertainty. To eliminate the influence of the variable parameters, neural networks are used for the calculation. For training the neural network, a set of possible values of the base resistances, loads, and the variable resistance of the multiport itself is specified. The corresponding set of input currents forms a training vector, and the load values are components of the target vector. Numerical experiments carried out in the Deep Learning package of the MATLAB system with one and two loads show the control calculation accuracy for the trained shallow neural network at a level of one to two percent.

Key words: multiport, load characteristic, wire line, projective transformations, cross ratio, neural network

REFERENCES

1. **Bessonov L.A.** *Teoreticheskiye osnovy elektrotehniki. Elektricheskiye tsepi* (Theoretical Foundations of Electrical Engineering. Electric Circuits). M.: Vysshaya shkola, 1996, 640 p.
2. **Bhattacharyya S.P., Keel L.H., Mohsenizadeh D.N.** *Linear Systems: A Measurement-Based Approach*. Springer, India, 2014, 89 p.
3. **Penin A.A.** *Proektivnaya geometriya v teorii elektricheskikh tsepey s peremennymi i nelineynymi nagruzkami: nauchnyy ref. dis. ... doktora tekhn. nauk.* (Projective Geometry in the Theory of Electric Circuits with Variable and Nonlinear Loads: Scientific Abstract of Dissertation of Dr. Sci. (Eng.)). Kishinev, 2017, 81 p.
4. **Penin A.** *Analysis of Electrical Circuits with Variable Load Regime Parameters: Projective Geometry Method*. Springer International Publishing, Cham, Switzerland, 2020, 520 p., DOI:10.1007/978-3-030-35366-7.
5. **Mazin V.D.** *Izmeritel'naya tekhnika – in Russ. (Measurement Techniques)*, 1980, No. 6, pp. 14–15.
6. **Mazin V.D.** *Error of Measurement in the Compound-Ratio Method. – Measurement Techniques*, 1983, vol. 26, No. 8, pp. 628–629, DOI: 10.1007/BF00833215.
7. **Tsybul'skiy O.A.** *Izmeritel'naya tekhnika – in Russ. (Measurement Techniques)*, 2013, No. 1, pp. 27–29.
8. **D'yakov V.P., Kruglov V.V.** *MATLAB 6.5 SP1/7/7 SP1/7 SP2+ Simulink 5/6. Instrumenty iskusstvennogo intellekta i bioinformatiki (MATLAB 6.5 SP1/7/7 SP1/7 SP2+ Simulink 5/6. Artificial Intelligence and Bioinformatics Tools)*. M.: SOLON-PRESS, 2009, 456 p.
9. **Burakov M.V.** *Neyronnye seti i neirokontrollery* (Neural Networks and Neurocontrollers). SPb.: GUAP, 2013, 284 p.
10. **Vakulenko S.A., Zhikhareva A.A.** *Prakticheskiy kurs po neyronnym setyam* (Practical Course on Neural Networks). SPb.: Universitet ITMO, 2018, 71 p.
11. **Deep Learning Toolbox** [Electron. resource], URL: <https://www.mathworks.com/help/deeplearning/index.html> (Date of appeal 20.02.2022).
12. **Glagolev N.A.** *Proektivnaya geometriya* (Projective geometry). M.: Vysshaya shkola, 1963, 343 p.
13. **Ayres F.** *Schaum's Outline Series Theory and Problems of Projective Geometry*. New York, McGraw–Hill, 1967, 249 p.
14. **Riaza R.** *Circuit theory in Projective Space and Homogeneous Circuit Models. – IEEE Transactions on Circuits and Systems I: Regular Papers*, 2019, vol. 66, No. 2, pp. 463–476.
15. **Sholle F.** *Glubokoe obuchenie na Python* (Deep Learning with Python). SPb.: Piter, 2018, 400 p.
16. **Novgorodtsev A.B.** *Raschet elektricheskikh tsepey v MATLAB* (Calculation of Electrical Circuits in MATLAB). SPb.: Piter, 2004, 250 p.
17. **Jin Z., Kaba, S.** *Deep Neural Network Based Behavioral Model of Nonlinear Circuits. – Journal of Applied Mathematics and Physics*, 2021, No. 9, pp. 403–412.

[20.02.2022]