

## СТАТЬИ

УДК 004.021:615.478:616-7

**ПРОБЛЕМЫ РЕАЛИЗАЦИИ АЛГОРИТМОВ РЕКОНСТРУКЦИИ МНОГОРАКУРСНОЙ ЭЛЕКТРОИМПЕДАНСНОЙ ТОМОГРАФИИ****Алексаян Г.К., Щербakov И.Д., Сулыз А.В., Тетелев М.В.***ФГБОУ ВО «Южно-Российский государственный политехнический университет (НПИ) им. М.И. Платова», Новочеркасск, e-mail: graer@yandex.ru*

Статья посвящена разработке алгоритма поиска числа входных данных для процесса выполнения дифференциальной реконструкции поля импеданса и визуализации её результатов методом многокурсовой электроимпедансной томографии. Исследованы зависимости нагрузки на персональный компьютер в составе информационно-измерительной системы в виде времени процесса реконструкции и частоты кадров в секунду от количества входных файлов, размера выделяемой оперативной памяти, тактовой частоты процессора для технических средств многокурсовой электроимпедансной томографии на базе персонального компьютера. Реконструкция и визуализация входных данных выполнялась в программном пакете *EIDORS* в среде программирования *Octave*. Разработан алгоритм поиска значения числа входных данных для одного томографического среза. Реализация данного алгоритма выполнена на языках программирования *Python* и *Java*. Данный алгоритм позволяет увеличить скорость обработки входных данных путём определения максимального количества входных файлов, прием которого программным компонентом аппаратно-программного комплекса электроимпедансной томографии возможен без превышения объема выделяемой оперативной памяти. При разработке данного алгоритма были проведены исследования на персональных компьютерах двух конфигураций по оценке средней нагрузки процесса реконструкции и числа файлов для процесса реконструкции. Проведенное исследование позволяет выбрать число входных данных для процесса реконструкции, как для уже полученных архивных данных, так и данных, получаемых в режиме реального времени. Данное исследование позволит увеличить скорость визуализации реконструированных изображений при минимальной задержке выполнения процесса реконструкции для двумерной и трехмерной электроимпедансной томографии.

**Ключевые слова:** электроимпедансная томография, многокурсовая электроимпедансная томография, алгоритмы реконструкции, дифференциальная реконструкция

**PROBLEMS OF IMPLEMENTATION OF MULTI-ANGLE ELECTRICAL IMPEDANCE TOMOGRAPHY RECONSTRUCTION ALGORITHMS****Aleksanyan G.K., Shcherbakov I.D., Sulyz A.V., Tetelev M.V.***Platov South-Russian State Polytechnic University (NPI), Novocherkassk, e-mail: graer@yandex.ru*

The article is devoted to the development of search algorithms the number of input data for the process of reconstruction of field Performing differential impedance and visualization of its results by multi-angle electrical impedance tomography. The dependence of the load on the PC in the information-measuring systems in terms of the time of the reconstruction process and the frame rate per second by the number of input files, the size of allocated memory, CPU clock speed for the technical means of the multi-electrical impedance tomography based on the PC. Reconstruction and visualization of input data was performed in *EIDORS* software packages in *Octave* programming environment. Developed search algorithm value of the number of input data for a tomographic slice. The implementation of this algorithm is executed in the *Python* programming language and *Java*. This algorithm allows to increase the speed of processing the input data by determining the maximum number of input files, and software components whose reception hardware and software complex electrical impedance tomography is possible without exceeding the scope of allocated RAM. In the development of this algorithm research on the personal computer of the two configurations were performed on rank average load of the reconstruction process and the number of files for the upgrade process. Conducting research to select the number of input data for the reconstruction process, both already received archival data, and the data obtained in real-time mode. This study will increase the speed of the rendering of images reconstructed with minimum delay Upgrade Process for two- and three-dimensional electrical impedance tomography.

**Keywords:** multi-angle electrical impedance tomography, algorithms, differential reconstruction

Электроимпедансная томография (ЭИТ) – неинвазивный метод реконструкции и визуализации распределения поля электрической проводимости внутренних структур биологических объектов по результатам электрических измерений на его поверхности [1–3]. В общем случае, практическая реализация метода ЭИТ заключается в подключении исследуемого объекта (ИО) к одному или нескольким источникам электрического тока

высокой частоты и малой амплитуды через электроды, размещенные на его поверхности, и регистрации потенциалов  $\varphi(i, j)$  в точках крепления электродов (где  $i$  – номер пары инжектирующих электродов,  $j$  – номер измерительного электрода) [1]. Используемые в ЭИТ алгоритмы реконструкции используют математическую модель объекта исследования, расчёт сопротивления объекта осуществляется путём решения дифферен-

циальных уравнений для каждого среза [2]. Одним из перспективных направлений ЭИТ является трехмерная ЭИТ, позволяющая получать информацию о параметрах поля проводимости внутренних структур ИО как в отдельном двухмерном томографическом срезе, так и в их совокупности [3]. Использование нескольких электродных поясов пациента позволяет формировать томографические срезы не только в плоскости электродных поясов, но и в других проекциях, предлагая новую функциональность в виде многоакракурности.

Цель исследования: реализация многоакракурной ЭИТ (МРЭИТ) в аппаратно-программных комплексах создает ряд проблем, одной из которых является ресурсоемкость процесса реконструкции.

Для определения параметров ресурсоемкости процесса дифференциальной реконструкции [2] (использованного объема оперативной памяти (ОЗУ) электронно-вычислительной машины (ЭВМ)  $V_{cp}$ , общего затраченного времени  $t_{общ}$ , времени формирования кадра  $t_{уд}$ , частоты кадров в секунду  $k$ ) проведено исследование, в котором использовано минимальное для дифференциальной реконструкции количество входных файлов  $n = 2$ , общее количество файлов  $N = 4260$ .

В качестве объекта исследования выбран фантом с цилиндрическими неоднородностями заданного размера [4]. Параметры ЭВМ, используемой для проведения исследования: размер установленного ОЗУ ЭВМ – 6 Гб, тактовая частота процессора ЭВМ – 3,20 ГГц, версия операционной системы (ОС) – *Windows 8*.

В результате проведенного исследования были получены следующие показатели ресурсозатратности:  $V_{cp} = 410,80$  Мб,  $t_{общ} = 513,30$  с,  $t_{уд} = 0,85$  с,  $k = 1,16$  кадров/с. Таким образом, выявлено, что при реконструкции поля проводимостей методом ЭИТ возникает проблема низкой скорости визуализации результатов реконструкции, не позволяющей при полученной  $k$  отслеживать динамику некоторых физиологических процессов (например, воздухонаполнение легких, сердцебиение).

Одним из путей решения данной проблемы является определение такого числа  $n$ , которое может обработать программный

пакет за один вызов процесса реконструкции без превышения  $V_{cp}$ . В связи с этим предложен для разработки алгоритм выбора необходимого количества  $n$  в зависимости от конфигурации используемого в составе аппаратно-программного комплекса персонального компьютера, заключающийся в тестировании компонентов программного обеспечения (ПО) для определения параметров обработки входных файлов для процесса реконструкции при используемой в задачах МРЭИТ конфигурации персонального компьютера.

### Материалы и методы исследования

Алгоритм реализован с помощью языка программирования *Python* [5]. Исследование разработанного алгоритма выполнено на АПК ЭИТ БО [6–8]. Процесс реконструкции и визуализации проводился с помощью программного пакета *EIDORS* [9]. Проведено исследование на двух ЭВМ, № 1 и № 2, с параметрами, приведенными на табл. 1, целью исследования являлось определение  $V_{cp}$  для процесса реконструкции одного томографического среза при МРЭИТ.

Блок-схема обобщенного алгоритма исследования  $V_{cp}$  приведена на рис. 1. Алгоритм работы программы, разработанной для проведения данного исследования, состоит в следующем: программой осуществляется прием заданного количества входных файлов  $n$  для процесса реконструкции, поэтапно обрабатываются все входные файлы  $N = 4260$ . После завершения процесса работы *EIDORS* в параметрах программы АПК ЭИТ БО задается новое значение  $n$  и регистрируются параметры  $V_{cp}$ ,  $t_{общ}$ ,  $t_{уд}$  и  $k$  для нового процесса реконструкции.

Результаты исследования зависимости  $V_{cp}$  от  $n$  приведены на табл. 2 и 3.

Анализ полученных результатов показывает, что при увеличении  $n$  для процесса реконструкции уменьшается  $V_{cp}$  и снижается время выполнения работы ПО *EIDORS*. При увеличении  $n$  до  $N$  зарегистрировано превышение объема выделяемой ОЗУ и завершение процесса реконструкции системной ошибкой. На рис. 2 и 3 приведены зависимости изменения  $V_{cp}$ ,  $t_{уд}$  процесса реконструкции одного файла от  $n$  для ЭВМ № 1 и ЭВМ № 2.

Таблица 1

Параметры исследуемых ЭВМ

	Версия операционной системы (ОС)	Размер ОЗУ	Тактовая частота процессора
№ 1	Windows 8	8 ГБ	3,5 ГЦ
№ 2	Windows 8	6 ГБ	3,2 ГЦ

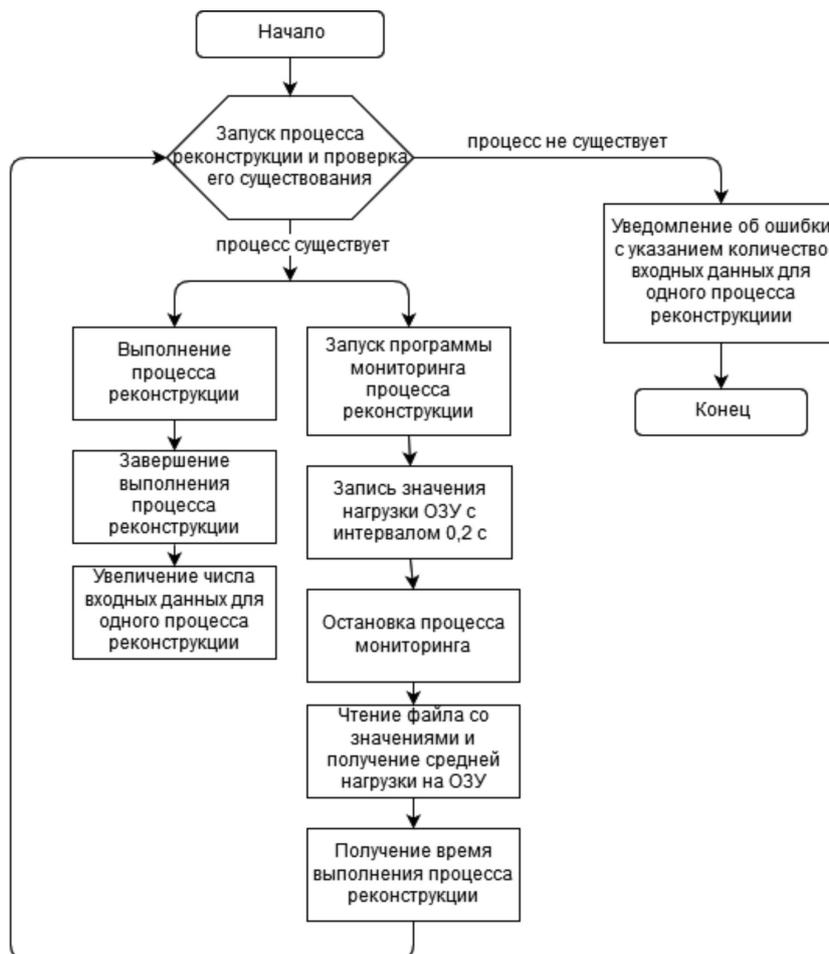


Рис. 1. Блок-схема обобщенного алгоритма исследования  $V_{cp}, t_{уд}, t_{общ}, k$

Таблица 2

Исследование зависимости  $V_{cp}, t_{уд}, t_{общ}, k$  от  $n$  для ЭВМ № 1

N	n	$V_{cp}$ , Мб	$t_{уд}$ , с	$t_{общ}$ , с	$k$ , кадров/с
4260	10	563,5	0,330	1283,70	3,3
4260	71	129,8	0,090	412,60	10,3
4260	355	84,8	0,083	354,70	12,0
4260	2130	83,5	0,081	346,80	12,3
4260	4260	–	–	–	–

Таблица 3

Исследование зависимости  $V_{cp}, t_{уд}, t_{общ}, k$  от  $n$  для ЭВМ № 2

N	n	$V_{cp}$ , Мб	$t_{уд}$ , с	$t_{общ}$ , с	$k$ , кадров/с
4260	10	549,9	3,77	16099,5	0,20
4260	71	125,6	0,45	1915,6	2,22
4260	355	81,4	0,31	1326,9	3,21
4260	2130	80,3	0,26	1121,3	3,79
4260	4260	–	–	–	–

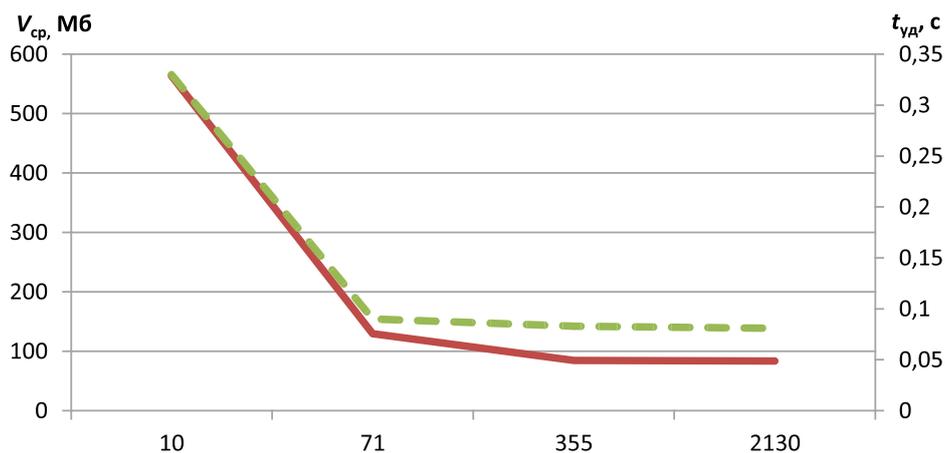


Рис. 2. Зависимость  $V_{cp}$ ,  $t_{уд}$  от  $n$  для ЭВМ № 1 в составе АПК ЭИТ БО

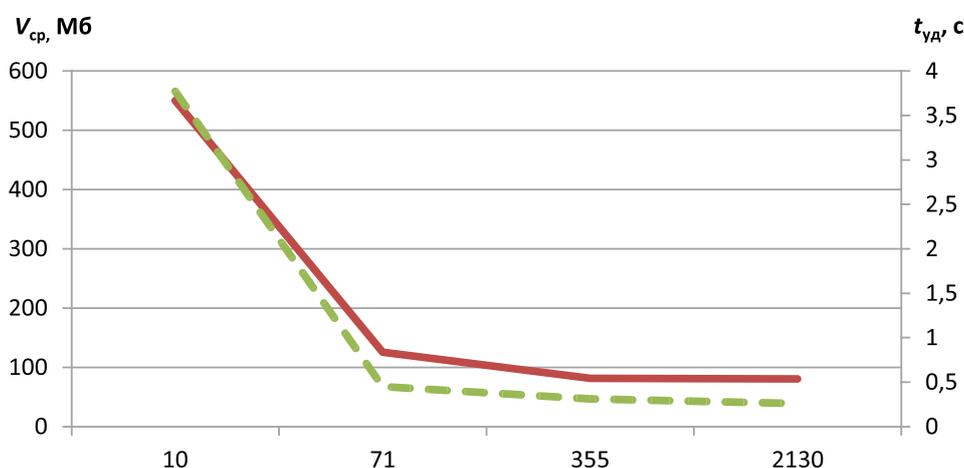


Рис. 3. Зависимость  $V_{cp}$ ,  $t_{уд}$  от  $n$  для ЭВМ № 2 в составе АПК ЭИТ БО

Сплошной линией обозначен график потребления  $V_{cp}$  процессом реконструкции, пунктирной линией обозначены связанные с его выполнением временные затраты  $t_{уд}$ . Анализ рис. 2 и 3 показывает, что значение показателей  $V_{cp}$  и  $t_{уд}$  процесса реконструкции уменьшается до достижения значения  $n = 71$  (1,66% от общего числа файлов), дальнейшее увеличение параметра  $n$  не приводит к значительному росту скорости обработки входных данных.

Для определения  $n$  для каждого процесса реконструкции, при котором ресурсоемкость его минимальна, разработан алгоритм, основанный на методе бинарного поиска.

Бинарный (двоичный) поиск – это поиск заданного элемента на упорядоченном множестве, осуществляемый путем неоднократного деления этого множества на две

части таким образом, что искомый элемент попадает в одну из этих частей. Поиск заканчивается при совпадении искомого элемента с элементом, который является границей между частями множества или при отсутствии искомого элемента [10].

Разработанный алгоритм позволяет определить максимальное  $n$  для одного процесса реконструкции без превышения  $V_{cp}$ . Данный алгоритм реализован на языке программирования *Python*. Блок-схема алгоритма представлена на рис. 4.

Как видно из представленной на рис. 4 блок-схемы, разработанный алгоритм может быть применен при каждом запуске процесса реконструкции, позволяя значительно сокращать связанные с ней временные затраты без риска возникновения аварийных ситуаций, связанных с переопределением ОЗУ.

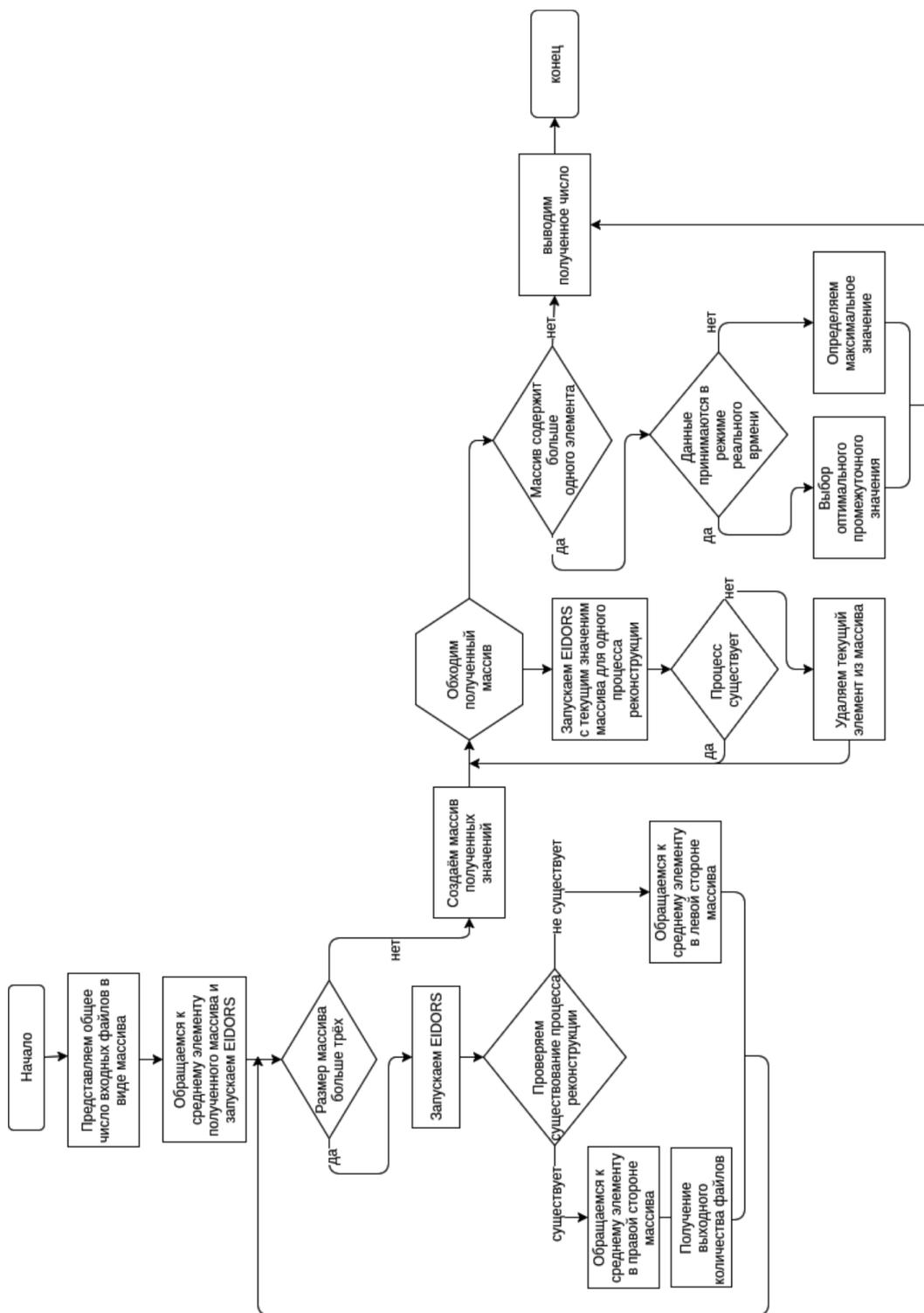


Рис. 4. Блок-схема работы алгоритма поиска n

### Результаты исследования и их обсуждение

Разработанный алгоритм позволяет определять максимальное  $n$  для достижения минимальной длительности процесса реконструкции и визуализации уже полученных входных файлов. При получении входных файлов в режиме реального времени (со скоростью 20–60 кадров в секунду [11]), производится выбор такого значения  $n$ , которое позволяет снизить время ожидания при получении входных файлов для процесса реконструкции и тем самым избежать увеличения времени задержки процесса выполнения реконструкции EIDORS и, соответственно, потери диагностической ценности реконструированных изображений.

### Заключение

Таким образом, в рамках проведенного исследования разработан алгоритм поиска значения числа входных данных для одного томографического среза. Программное обеспечение, реализующее данный алгоритм, позволяет увеличить скорость обработки входных данных путём определения максимального количества входных файлов, прием которого программным компонентом АПК ЭИТ БО на имеющейся конфигурации ЭВМ возможен без риска превышения объема выделяемой ОЗУ и, как следствие, аварийного завершения работы процесса реконструкции.

При разработке данного алгоритма были проведены исследования на ЭВМ двух пространственных конфигураций по оценке средней нагрузки процесса реконструкции и числа файлов для процесса реконструкции, которое позволит значительно увеличить скорость визуализации реконструированных изображений при сохранении допустимого показателя временной задержки, связанной с выполнением процесса EIDORS как для двухмерной, так и трехмерной ЭИТ.

Проведенное исследование позволяет определить число входных данных для процесса реконструкции, как для уже полученных архивных данных, так и данных, получаемых в режиме реального времени в задачах МРЭИТ.

Направлением дальнейших исследований является оптимизация используемых алгоритмов реконструкции для задач МРЭИТ. Полученные результаты найдут применение в последующих исследованиях по разработке технических средств двухмерной и трехмерной ЭИТ.

*Работы выполняются в рамках гранта Президента Российской Федерации для государственной поддержки молодых российских ученых МК-196.2017.8 «Разработка теоретических основ и алгоритмов многоакурсной электроимпедансной томографии для систем неинвазивной трехмерной медицинской визуализации».*

### Список литературы

1. Пеккер Я.С., Бразовский К.С., Усов В.Ю., Плотников М.П., Уманский О.С. Электроимпедансная томография. Томск: Издательство «НТЛ», 2004. 192 с.
2. Коломиец В.Я. Электроимпедансная томография новый метод респираторного мониторинга // Поликлиника. 2013. С. 34–35.
3. Yang C.L. Electrical impedance tomography: algorithms and applications: Thesis (Doctor of Philosophy (PhD)). University of Bath. 2014. P. 142.
4. Aleksanyan G., Shcherbakov I., Kucher A., Priyma M. Research of the conductivity of organic and inorganic media in multi-angle multi-frequency electrical impedance tomography. MATEC Web Conf. № 04008. 2017. P. 1–4.
5. Python 3. 2016. [Электронный ресурс]. URL: <https://www.python.org/> (дата обращения: 22.10.2019).
6. Алексанян Г.К., Кучер А.И., Горбатенко Н.И., Нгуен Мань Кыонг, Чан Нам Фонг. Разработка информационно-измерительной системы электроимпедансной томографии биологических объектов // Вьетнамо-российская Междунар. науч. конф., ГТУ им. Ле Куи Дона, 02–03 апр. 2015 г.: тез. докл. / ГТУ им. Ле Куи Дона. Ханой: ЛКД, 2015. С. 52–53.
7. Aleksanyan G., Gorbatenko N., Kucher A., Shcherbakov I. Hardware in the loop simulation of objects internal structures inhomogeneities in multi-angle electrical impedance tomography. MATEC Web Conf. № 04007. 2017. P. 1–4.
8. Aleksanyan G.K., Shcherbakov I.D., Kucher A.I. Feature research of using current source in 2-dimensional and 3-dimensional multifrequency electrical impedance tomography devices. Journal of Engineering and Applied Sciences. 2017. Vol. 12. № 3. P. 587–592.
9. EIDORS: Electrical Impedance Tomography and Diffuse Optical Tomography Reconstruction Software. 2016. [Electronic resource]. URL: <http://eidors3d.sourceforge.net> (date of access: 22.10.2019).
10. Левитин А. Введение в разработку и анализ. М.: Вильямс, 2006. С. 180–183.
11. Emamjomeh-Zadeh E., Kempe D., Singhal. Deterministic and probabilistic binary search in graphs. 48th ACM Symposium on Theory of Computing. 2016. P. 519–532.
12. Teschner E., Imhoff M., Leonhardt S. Electrical Impedance Tomography: The realisation of regional ventilation monitoring. 2nd edition. Drägerwerk AG & Co, 2015. P. 36–37.