



<http://dx.doi.org/10.35596/1729-7648-2022-28-4-72-79>

Оригинальная статья  
Original paper

УДК 004.021+004.023+004.514+004.932

## ИНТЕРАКТИВНАЯ ВИЗУАЛИЗАЦИЯ АЛГОРИТМОВ ТРАССИРОВКИ ПЕЧАТНЫХ ПРОВОДНИКОВ В УЧЕБНОМ ПРОЦЕССЕ

Д. О. ПЕТРОВ

*Брестский государственный технический университет (г. Брест, Республика Беларусь)*

*Поступила в редакцию 29.06.2022*

© Белорусский государственный университет информатики и радиоэлектроники, 2022

**Аннотация.** Разработан программный модуль, позволяющий производить интерактивную демонстрацию лабиринтных алгоритмов трассировки печатных проводников, которые при своей работе используют сеточное представление дискретного рабочего пространства печатной платы. Основным отличием разработанного модуля от программного обеспечения аналогичного назначения является возможность пошагового аннотированного выполнения различных алгоритмов трассировки печатных проводников с возможностью изменения порядка проведения заданных пользователем трасс. Рассмотрены взаимосвязь алгоритмов нахождения кратчайших путей на графах и алгоритмов трассировки печатных проводников и влияние очередности проведения трасс между парами контактных площадок на длину печатных электрических соединений. Проведен анализ достоинств и недостатков волнового алгоритма (алгоритма Ли) и эвристического алгоритма A\* с описанием их пошаговой работы. Применение разработки оправдано при чтении лекций и проведении лабораторных работ по теоретическим основам систем автоматизации проектирования радиоэлектронной аппаратуры и способствует решению комплексной задачи цифровизации образовательного процесса при помощи программ-визуализаторов, позволяющих наблюдать результаты работы реализуемых ими алгоритмов на различных наборах исходных данных.

**Ключевые слова:** цифровизация образования, визуализация алгоритма, система автоматизации проектирования, печатная плата, трассировка, дискретное рабочее пространство.

**Конфликт интересов.** Автор заявляет об отсутствии конфликта интересов.

**Для цитирования.** Петров Д. О. Интерактивная визуализация алгоритмов трассировки печатных проводников в учебном процессе. *Цифровая трансформация*. 2022; 28 (4): 72–79.

## INTERACTIVE VISUALIZATION OF THE PRINTED CIRCUITS TRACING ALGORITHMS FOR EDUCATIONAL PURPOSES

DMITRIY O. PETROV

*Brest State Technical University (Brest, Republic of Belarus)*

*Submitted 29.06.2022*

© Belarusian State University of Informatics and Radioelectronics, 2022

**Abstract.** A software module that allows interactive demonstration of maze routing algorithms for tracing printed conductors, which, during their operation, use a grid representation of a discrete working space of a printed circuit board has been developed. The main difference between the developed module and the software of similar purpose is the possibility of step-by-step annotated execution of various algorithms for tracing printed conductors with the possibility of changing the order of routing user-specified conductors. The interrelation of algorithms for finding the shortest paths on graphs and algorithms for tracing printed conductors and the influence of the order of making connections between pairs of contact pads on the length of printed electrical connections are considered.

An analysis of the advantages and disadvantages of the wave algorithm (Lee algorithm) and the heuristic algorithm A\* was carried out with a description of their step-by-step operation. The use of the development is justified when giving lectures and conducting laboratory research on the theoretical foundations of automation systems for the design of radio electronic equipment and contributes to solving the complex problem of digitalization of the educational process with the help of visualization programs that allow you to observe the results of the algorithms they implement on various sets of initial data.

**Keywords:** digitalization of education, visualization of algorithm, design automation system, printed circuit board, tracing, discrete workspace.

**Conflict of interests.** The author declares no conflict of interests.

**For citation.** Petrov D. O. Interactive Visualization of the Printed Circuits Tracing Algorithms for Educational Purposes. *Digital Transformation*. 2022; 28 (4): 72–79.

## Введение

При изучении технических дисциплин, тесно связанных с информационными технологиями, перед преподавателем встает проблема наглядного изображения процесса работы различных алгоритмов для повышения степени усвоения учебного материала студентами. Применение современных технических средств обучения при чтении лекций позволяет отображать не только статичную и динамическую инфографику, но и привнести элементы интерактивной демонстрации использованием специализированных программ-визуализаторов. Программы-визуализаторы предоставляют возможность наблюдать в пошаговом режиме результаты работы реализуемых ими алгоритмов на различных наборах исходных данных, упрощают взаимодействие преподавателя с аудиторией при поступлении вопросов со стороны учащихся и способствуют улучшению понимания ими изучаемого материала [1–3].

Одна из сложных задач, решаемых при автоматизации разработки радиоэлектронной аппаратуры, – проектирование печатного монтажа. Эта задача состоит в трассировке электрических проводников между множеством пар контактных площадок согласно принципиальной электрической схеме электронного устройства. Обучение студентов теоретическим основам систем автоматизации проектирования радиоэлектронной аппаратуры предполагает знакомство с группой алгоритмов трассировки печатных проводников, относящихся к классу лабиринтных (maze router) [4], которые при своей работе используют сеточное представление дискретного рабочего пространства (ДРП) печатной платы [5]. На практике дискретное рабочее пространство – это двумерная матрица, состоящая из элементов квадратной формы, центры которых представляют собой вершины ортогонального связного неориентированного графа, при этом трассировка печатных проводников соответствует поиску пути в таком графе.

## Классические лабиринтные алгоритмы трассировки печатных проводников

### Алгоритм Ли

Важное место среди лабиринтных алгоритмов трассировки занимает алгоритм Ли [6, 7], разработанный в 1961 году и фактически реализующий поиск в ширину в графе, предложенный в 1959-м Э. Ф. Муром [8]. Алгоритм гарантирует построение кратчайшей трассы между двумя контактными площадками при существовании пути между ними на ДРП печатной платы. Дискретное рабочее пространство представляет собой прямоугольную матрицу  $M$ , состоящую из элементов квадратной формы, содержащую  $m$  строк и  $n$  столбцов. С каждым из элементов матрицы  $M$  связаны два атрибута:  $f$  – признак наличия препятствия проведению трассы и  $l$  – метка, соответствующая номеру текущей итерации поиска кратчайшего пути между контактными площадками. Начальной  $S$  и конечной  $F$  контактными площадкам соответствуют два различных элемента матрицы  $M$ . При работе алгоритма используются три множества ячеек ДРП, обозначенные как  $V$ ,  $W$ ,  $T$ , и переменная  $w$ .

Словесное описание алгоритма Ли представим следующим образом.

1. В соответствии с проектом печатной платы определяется размер  $m \times n$  матрицы  $M$ . У элементов матрицы ДРП, через которые разрешено проводить трассы, атрибутам  $f$  присваивается значение, равное нулю (в дальнейшем такие элементы называются свободными), в противном случае при наличии препятствий проведению трассы атрибутам  $f$  присваивается значение,

равное единице. Атрибутам  $l$  всех элементов матрицы присваивается значение, равное  $\infty$ . Переменная  $w$  и атрибут  $l$  элемента ДРП, соответствующего начальной контактной площадке  $S$ , принимают нулевое значение.  $V := \emptyset$ ,  $W := \emptyset$ ,  $T := \emptyset$ .

2. В множество  $V$  вносятся все свободные элементы матрицы со значением атрибута  $l$ , равным нулю.

3. В множество  $W$  вносятся все свободные элементы ДРП со значением атрибута  $l = \infty$ , имеющие в окрестности фон Неймана входящие в множество  $V$  элементы.

4. Если  $W = \emptyset$ , то проведение трассы между контактными площадками  $S$  и  $F$  невозможно, и следует перейти к шагу 9.

5.  $w := w + 1$ . Атрибутам  $l$  всех элементов, входящих в множество  $W$ , присваивается значение переменной  $w$ .

6. Если элемент матрицы  $M$ , соответствующий конечной контактной площадке  $F$ , входит в множество  $W$ , тогда необходимо перейти к шагу 8.

7.  $V := W$ ,  $W := \emptyset$ . Переход к шагу 3.

8. Последовательно включить в множество  $T$  элементы ДРП с уменьшающимися на единицу значениями атрибута  $l$ , начиная с конечной контактной площадки  $F$  ( $l = w$ ) и заканчивая  $S$  ( $l = 0$ ).

9. Множество  $T$  содержит элементы ДРП, представляющие собой трассу, проведенную между  $S$  и  $F$ .

#### Алгоритм $A^*$

Несмотря на гарантированное отыскание существующего кратчайшего пути между контактными площадками, на быстроедействие алгоритма Ли отрицательно влияет высокая вычислительная сложность, характерная для поиска в ширину в неориентированном графе. Вычислительную сложность поиска в ширину возможно сократить, снижая количество обследуемых вершин графа путем информированного их отбора на основе эвристического подхода, как это описано в алгоритме  $A^*$  [9], разработанном в 1968 году для поиска кратчайшего пути между вершинами связанного графа и часто находящим свое применение при трассировке печатных проводников [10, 11].

Исходными для работы алгоритма  $A^*$  являются следующие данные: дискретное рабочее пространство в виде прямоугольной матрицы  $M$ , состоящей из элементов квадратной формы, содержащей  $m$  строк и  $n$  столбцов; начальная  $S$  и конечная  $F$  контактные площадки, которым соответствуют два различных элемента матрицы  $M$ . С каждым из элементов матрицы  $M$  связаны четыре атрибута:  $f$  – признак наличия препятствия проведению трассы,  $g$  – длина пути от начальной контактной площадки до элемента,  $h$  – эвристическая оценка длины пути от элемента до конечной контактной площадки,  $l$  – ссылка на предшествующий по пути элемент. В качестве эвристической оценки длины пути между ячейками матрицы в данном случае рекомендуется применять так называемое манхэттенское расстояние, т. е. расстояние между двумя точками равно сумме модулей разностей их координат по строкам и столбцам матрицы  $M$ . При работе алгоритма используются три множества элементов ДРП, обозначенные как  $P$ ,  $Q$ ,  $T$ , переменные  $x$ ,  $z$ , соответствующие обрабатываемым элементам матрицы дискретного рабочего пространства модели печатной платы, и представляющая собой ссылку на элемент ДРП переменная  $v$ . Для сокращения словесного описания алгоритма при обращении к атрибуту поименованного элемента матрицы ДРП будет использована точечная нотация, например:  $x.g \equiv$  атрибут  $g$  элемента матрицы  $M$  с именем  $x$ .

Словесное описание алгоритма  $A^*$  представим следующим образом.

1. В соответствии с проектом печатной платы определяется размер  $m \times n$  матрицы  $M$ . У элементов матрицы, через которые разрешено проводить трассы, атрибутам  $f$  присваивается значение, равное нулю (в дальнейшем такие элементы называются свободными), в противном случае при наличии препятствий проведению трассы атрибутам  $f$  присваивается значение, равное единице.  $P := \emptyset$ ,  $Q := \emptyset$ ,  $T := \emptyset$ . Атрибутам элемента, соответствующего стартовой контактной площадке  $S$ , присваиваются следующие значения:  $g := 0$ ,  $h :=$  манхэттенское расстояние между стартовой и конечной контактными площадками,  $l :=$  нет значения.

2. В множество  $P$  включается элемент ДРП, соответствующий стартовой контактной площадке  $S$ .

3. Если  $P = \emptyset$ , то пути между контактными площадками  $S$  и  $F$  не существует, и следует перейти к шагу 9.

4. Элемент  $x$  матрицы ДРП с минимальным значением суммы атрибутов  $g$  и  $h$  (в дальнейшем такой элемент ДРП будет называться элементом с минимальной стоимостью), находящийся в множестве  $P$ , перемещается из множества  $P$  в множество  $Q$ , а ссылка на него присваивается переменной  $v$ . Если  $x \equiv F$ , тогда следует перейти к шагу 7.

5. Рассматриваются все элементы ДРП, находящиеся в окрестности фон Неймана относительно элемента с минимальной стоимостью и не принадлежащего множеству  $Q$ . Для каждого элемента  $z$  ДРП, удовлетворяющего указанным условиям, выполняются следующие действия:

5.1. если  $z \in P$  и величина  $z.g > x.g + 1$ , то  $z.g := x.g + 1$ , а  $z.l := v$ ;

5.2. если  $z \notin P$ , то  $z.l := v$ ,  $z.g := x.g + 1$ ,  $z.h :=$  манхэттенское расстояние между  $z$  и конечной контактной площадкой, включить  $z$  в множество  $P$ .

6. Переход к шагу 3.

7. Включить элемент  $x$  матрицы ДРП в множество  $T$ ,  $z := xl$ .

8. Если  $z \neq$  нет значения, то следует выполнять следующие действия:

8.1. включить элемент  $z$  в множество  $T$ ,  $z := z.l$ ;

8.2. переход к шагу 8.

9. Множество  $T$  содержит элементы ДРП, представляющие собой трассу, проведенную между  $S$  и  $F$ .

### Особенности последовательного построения множества трасс

Очередность построения множества соединений между контактными площадками в значительной степени влияет как на качество трассировки, так и на саму возможность провести все трассы. На рис. 1 изображена ситуация, когда проведение трассы между парой контактных площадок С–D делает невозможным выполнение соединения между площадками E и F.

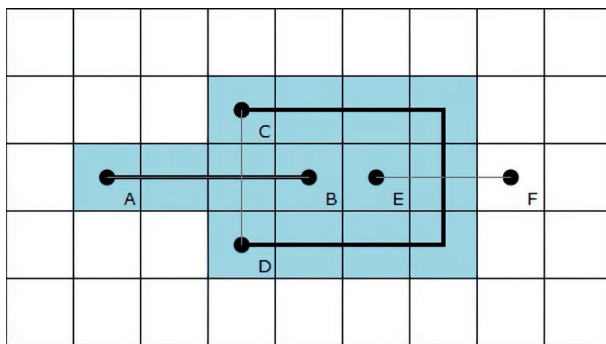


Рис. 1. Пример конфликта при построении трасс С–D и E–F  
Fig. 1. An example of a conflict in the construction of C–D and E–F wires

Устранение возникшей проблемы возможно при изменении очередности трассировки соединений: сначала выполняется соединение E–F, а затем С–D, как показано на рис. 2. Также порядок трассировки способен повлиять и на длину соединений.

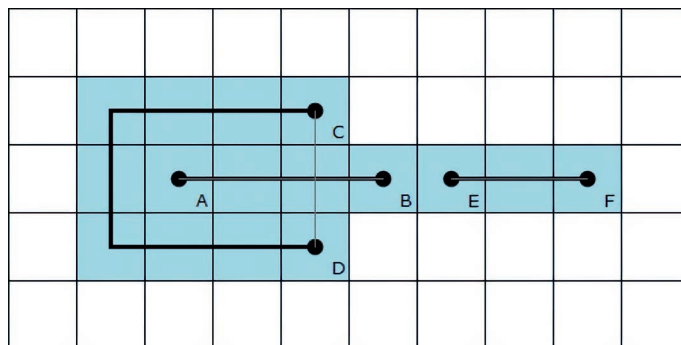


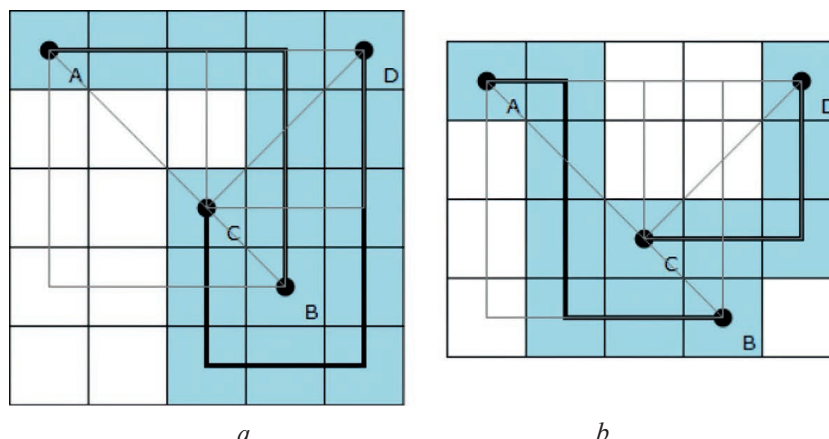
Рис. 2. Пример устранения конфликта при построении трасс С–D и E–F  
Fig. 2. An example of conflict resolution when constructing C–D and E–F wires

На рис. 3, *a* можно видеть неоправданное увеличение длины трассы С–D при выполнении соединения сначала между парой площадок А–В, а затем между парой С–D. Изменение очередности прокладки трасс устраняет выявленный недостаток (рис. 3, *b*). Один из способов определения корректной очередности прокладки трасс – учет количества контактных площадок внутри прямоугольных областей, противоположными углами которых являются конечные точки соответствующих соединений, при этом трассировка выполняется в порядке убывания отмеченного количества [12]. Например, на рис. 3 в прямоугольную область А–В попадает одна контактная площадка С, а область С–D не содержит ни одной площадки, следовательно, сначала прокладывается соединение С–D, а затем А–В. Необходимо отметить, что указанный способ упорядочивания соединений не работает в случае, изображенном на рис. 1, 2.

### Разработка визуализатора алгоритмов лабиринтной трассировки

Программный модуль визуализации алгоритмов трассировки представляет собой приложение, пользовательский интерфейс которого имеет вид классического диалогового окна.

На рис. 4 изображено главное окно разработанного приложения, а основные его элементы пронумерованы следующим образом: 1 – выпадающий список выбора демонстрируемого алгоритма трассировки; 2 – переключатель последовательных этапов подготовки исходных данных (установка препятствий, задание соединяемых пар контактных площадок) и начала выполнения трассировки; 3 – переключатель автоматической оптимизации порядка прокладки соединений, доступный на этапе задания трасс; 4 – группа кнопок, управляющая выполнением трассировки при заданных исходных данных (пошаговое выполнение алгоритма и построение всех трасс с предъяснением окончательного результата); 5 – основная рабочая область визуализатора алгоритмов; 6 – список последовательности прокладки заданных трасс; 7 – группа кнопок для корректировки списка последовательности прокладки заданных трасс (удаление из списка выделенной трассы и обмен местами двух выделенных трасс доступны на этапе задания соединяемых пар контактных площадок).



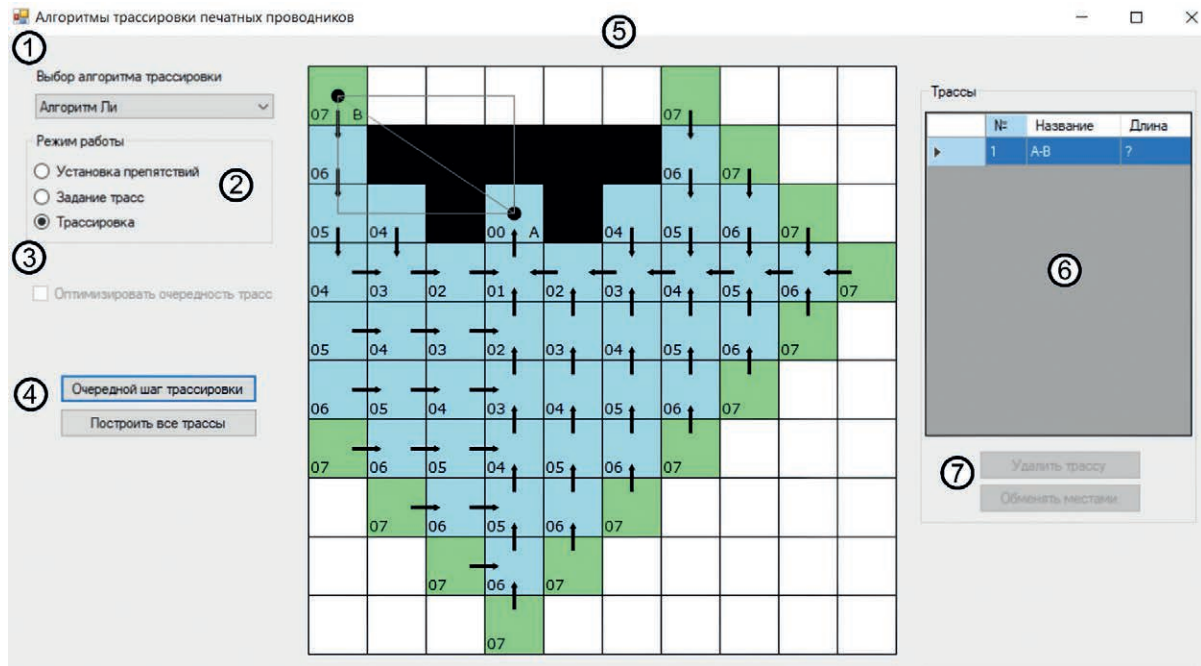
**Рис. 3.** Пример изменения длины трассы С–D в зависимости от очередности трассировки соединений А–В и С–D: *a* – сначала А–В, затем С–D; *b* – сначала С–D, затем А–В

**Fig. 3.** An example of C–D wire length difference, depending on the order of A–B and C–D routing: *a* – A–B first, C–D next; *b* – C–D first, A–B next

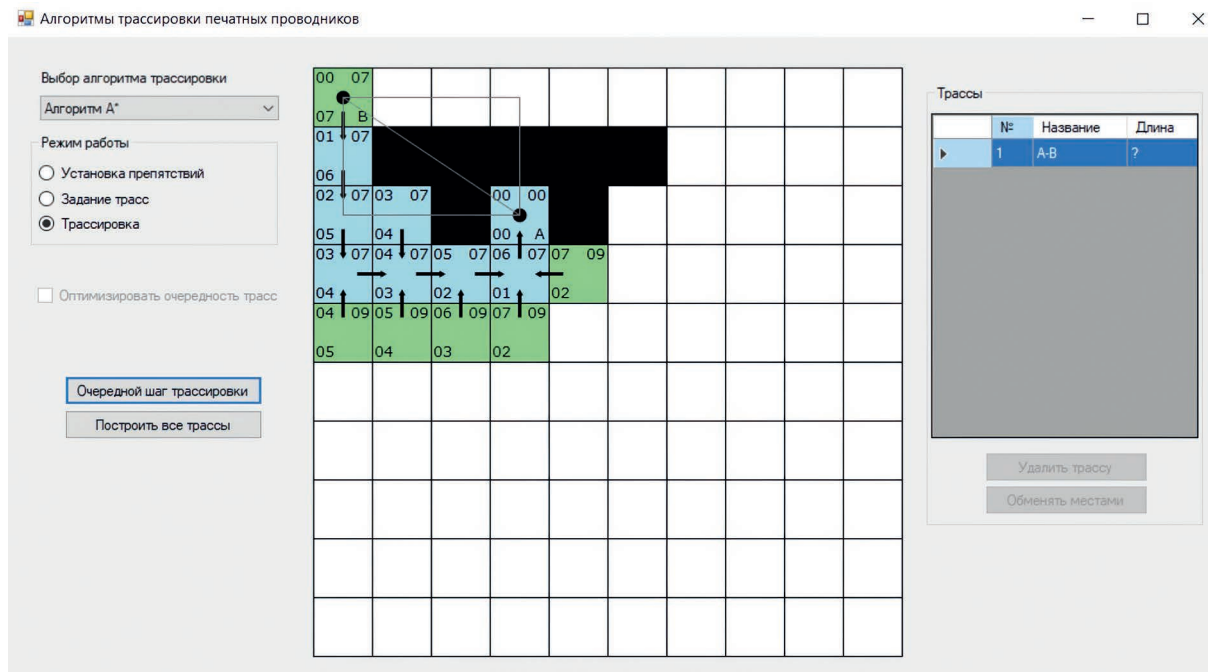
При демонстрации алгоритма Ли (рис. 4) выполняется нумерация элементов ДРП в зависимости от исполняемого шага, цветом выделяются пограничные элементы рабочего пространства, входящие на текущем этапе алгоритма в множество  $W$ , и стрелками обозначаются их непосредственные соседи из множества  $V$ .

Демонстрация алгоритма  $A^*$  (рис. 5) наглядно показывает значительно меньшее количество обследуемых элементов ДРП по сравнению с алгоритмом Ли (рис. 4). На элементы рабочего пространства наложены значения атрибутов  $g$  (левый нижний угол),  $h$  (левый верхний угол), а также сумма их величин (правый верхний угол). Стрелками обозначены предыдущие по пути элементы, соответствующие атрибуту  $l$ .





**Рис. 4.** Графический интерфейс пользователя разработанного программного модуля визуализации алгоритмов трассировки печатных проводников  
**Fig. 4.** Graphical user interface of the developed software module for visualization of maze routing algorithms



**Рис. 5.** Пример трассировки печатного проводника алгоритмом A\*  
**Fig. 5.** An example of wire routing using the A\* algorithm

При пошаговой демонстрации алгоритмов в момент достижения конечной контактной площадки аннотированно демонстрируется построение кратчайшего пути к начальной контактной площадке, а затем предьявляется конечный результат прокладки трассы.

На примере алгоритма A\* рис. 6, а представляет собой аннотацию построения пути от конечной контактной площадки до начальной, а рис. 6, b изображает результат прокладки трассы.

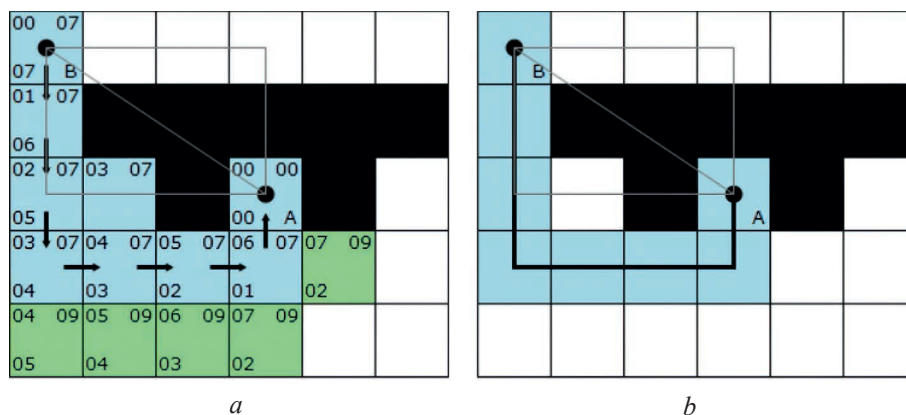


Рис. 6. Пример процесса оформления прокладки трассы: *a* – аннотация; *b* – конечный результат  
Fig. 6. An example of annotated wire routing: *a* – annotation; *b* – final result

### Заключение

Разработанный программный модуль визуализации алгоритмов трассировки проходит опытную эксплуатацию в Брестском государственном техническом университете при чтении лекций по дисциплинам «Системы автоматизированного проектирования» и «Автоматизация проектирования вычислительных систем» для студентов четвертого курса специальностей 1-39 03 02 «Программируемые мобильные системы», 1-36 04 02 «Промышленная электроника» и 1-40 02 01 «Вычислительные машины, системы и сети», что способствует решению комплексной задачи цифровизации образовательного процесса. Очевидна необходимость расширения программного модуля визуализаторами других известных алгоритмов трассировки печатных проводников, что при условии открытости исходного кода может быть выполнено самими студентами для повышения их квалификации и степени усвоения учебного материала в рамках проводимого лабораторного практикума по указанным выше предметам.

### Список литературы

1. Моглан, Д. Дидактический потенциал использования систем визуализации алгоритмов в процессе обучения программированию / Д. Моглан // Открытое образование. 2019. Т. 23, № 2. С. 31–41.
2. Цехан, О. Б. Обучающая программа – визуализатор алгоритма направленного перебора по векторной решетке / О. Б. Цехан // Информатизация образования – 2012: педагогические основы разработки и использования электронных образовательных ресурсов: матер. Междунар. науч. конф., Минск, 24–27 окт. 2012 г. Минск: БГУ, 2012. С. 79–83.
3. Казаков, М. А. Разработка логики визуализаторов алгоритмов на основе конечных автоматов / М. А. Казаков, Г. А. Корнеев, А. А. Шалыто // Телекоммуникации и информатизация образования. 2003. № 6. С. 27–58.
4. Веренич, И. Ю. Планарные и координатные трассировщики на практике / И. Ю. Веренич, Ю. В. Лысенко // Вестник ЮУрГУ. 2011. № 2. С. 30–33.
5. Бершадский, А. М. Алгоритм улучшения трассировки печатных плат / А. М. Бершадский, П. А. Гудков, Е. М. Подмарькова // Модели, системы, сети в экономике, технике, природе и обществе. 2021. № 3. С. 80–90.
6. Lee, C. Y. An Algorithm for Path Connections and its Applications / C. Y. Lee // IRE Transactions on Electronic Computers. 1961. No 10. P. 346–365.
7. Rubin, F. The Lee Path Connection Algorithm / F. Rubin // IEEE Transactions on Computers. 1974. Vol. C-23, No 9. P. 907–914.
8. Moore, E. F. The Shortest Path Through a Maze / E. F. Moore // Proc. International Symposium on the Theory of Switching. USA: Harvard University Press, 1959. P. 285–292.
9. Hart, P. E. A Formal Basis for the Heuristic Determination of Minimum Cost Path / P. E. Hart, N. J. Nilsson, B. Raphael // IEEE Transactions on Systems Science and Cybernetics. 1968. Vol. 4, No 2. P. 100–107.
10. Clow, G. W. A Global Routing Algorithm for General Cells / G. W. Clow // 21<sup>st</sup> Design Automation Conference Proceedings, June 25, 1984. P. 45–51.
11. Yuan, C. ASA-routing: a-Star Adaptive Routing Algorithm for Network-on-Chips / C. Yuan, J. Xiang // 18<sup>th</sup> International Conference, ICA3PP 2018, Proceedings, Part II, Guangzhou, China, Nov. 15–17, 2018. P. 187–198.
12. Chen, Huang-Yu. Global and Detailed Routing / Huang-Yu Chen, Yao-Wen Chang // Electronic Design Automation. Boston: Morgan Kaufmann, 2009. Chapter 12. P. 687–749.

## References

1. Moglan D. (2019) The Didactic Potential of Using Algorithm Visualization Systems in the Process of Programming Teaching. *Otkrytoe Obrazovanie* [Open Education]. 23 (2), 31–41 (in Russian).
2. Tsehan O. B. (2012) Educational Visualizer Software for Directed Search Algorithm on Vector Lattice. *Informatizaciya Obrazovaniya – 2012: Pedagogicheskie Osnovy Razrabotki i Ispol'zovaniya Elektronnyh Obrazovatel'nyh Resursov: Mater. Mezhdunar. Nauch. Konf.* [Informatization of Education – 2012: Pedagogical Foundations for the Development and Use of Electronic Educational Resources: Materials of International Scientific Conference]. Minsk, Oct. 24–27, 2012. Minsk, Belarussian State University. 79–83 (in Russian).
3. Kazakov M. A., Korneev G. A., Shalyto A. A. (2003) Development of the Logic of Algorithms Visualizers Based on Finite State Machines. *Telekommunikacii i Informatizaciya Obrazovaniya* [Telecommunications and Informatization of Education]. (6), 27–58 (in Russian).
4. Verenich I. U., Lysenko U. V. (2011) Planar and Coordinate Tracers in Practice. *Vestnik Uzhno-Ural'skogo Gosudarstvennogo Universiteta* [Bulletin of South Ural State University]. (2), 30–33 (in Russian).
5. Bershadjskij A. M., Gudkov P. A., Podmar'kova E. M. (2021) PCB Routing Improvement Algorithm. *Modeli, Sistemy, Seti v Ekonomike, Tekhnike, Prirode i Obshchestve* [Models, Systems, Networks in Economics, Technology, Nature and Society]. (3), 80–90 (in Russian).
6. Lee C. Y. (1961) An Algorithm for Path Connections and its Applications. *IRE Transactions on Electronic Computers*. (10), 346–365.
7. Rubin F. (1974) The Lee Path Connection Algorithm. *IEEE Transactions on Computers*. C-23 (9), 907–914.
8. Moore E. F. (1959) The Shortest Path Through a Maze. *Proc. International Symposium on the Theory of Switching*. USA, Harvard University Press. 285–292.
9. Hart P. E., Nilsson N. J., Raphael B. (1968) A Formal Basis for the Heuristic Determination of Minimum Cost Path. *IEEE Transactions on Systems Science and Cybernetics*. 4 (2), 100–107.
10. Clow G. W. (1984) A Global Routing Algorithm for General Cells. *21<sup>st</sup> Design Automation Conference Proceedings, June 25*. 45–51.
11. Yuan C., Xiang J. (2018) ASA-Routing: a-Star Adaptive Routing Algorithm for Network-on-Chips. *18<sup>th</sup> International Conference, ICA3PP 2018, Proceedings, Part II, Guangzhou, China, Nov. 15–17*. 187–198.
12. Chen Huang-Yu, Chang Yao-Wen (2009) Global and Detailed Routing. *Electronic Design Automation*. Boston, Morgan Kaufmann. Chapter 12. 687–749.

## Сведения об авторе

**Петров Д. О.**, к. т. н., доцент кафедры ЭВМ и системы Брестского государственного технического университета.

## Адрес для корреспонденции

224017, Республика Беларусь,  
г. Брест, ул. Московская, 267  
Брестский государственный  
технический университет  
Тел. +375 29 523-87-23  
E-mail: polegdo@gmail.com  
Петров Дмитрий Олегович

## Information about the author

**Petrov D. O.**, Cand. of Sci., Associate Professor of the Computer and Computer Sciences Department of Brest State Technical University.

## Address for correspondence

224017, Republic of Belarus,  
Brest, Moskovskaya St., 267  
Brest State Technical University  
Tel. +375 29 523-87-23  
E-mail: polegdo@gmail.com  
Petrov Dmitriy Olegovich