

УДК 658.52.011.56

## РАЗБИЕНИЕ РАЗВЕРТКИ 3D ГЕОМЕТРИЧЕСКОГО ОБЪЕКТА НА ПОЛИГОНАЛЬНЫЕ ПРИМИТИВЫ НА ПРИМЕРЕ КУБА, ОКТАЭДРА И КУРНОСОГО ДОДЕКАЭДРА

С. А. Зыкин<sup>1</sup>, А. А. Петренко<sup>2</sup>, Ю. И. Валиахметова<sup>3</sup>

<sup>1</sup>zykin\_sergey@mail.ru, <sup>2</sup>hawk321@mail.ru, <sup>3</sup>julikas@inbox.ru

<sup>1</sup> Лысьвенский филиал ФГБОУ ВО «Пермский национальный исследовательский политехнический университет» (ЛФ ПНИПУ)

<sup>2</sup> ФГБОУ ВО «Пермский национальный исследовательский политехнический университет» (ПНИПУ)

<sup>3</sup> ФГБОУ ВО «Уфимский государственный авиационный технический университет» (УГАТУ)

*Поступила в редакцию 09.11.2020*

**Аннотация.** При изготовлении заготовок необходимо решить задачу оптимального раскроя. Для этого выбирается один из наиболее выгодных методов раскроя материала для уменьшения количества отходов при нарезке заготовок. Необходимо расположить определенные объекты на поверхности другого, как правило, более крупного объекта так, чтобы площадь покрываемого объекта была заполнена целиком, без перекрытия между собой покрывающих объектов. В статье рассмотрен пример рационального раскроя разверток 3D-объектов с использованием трех подходов полигонального покрытия. Полученные результаты позволяют увидеть карту рационального покрытия, полученную тремя алгоритмами. Практическая значимость работы обусловлена значительным повышением роли рационального покрытия материала в связи с целесообразностью минимизации отходов.

**Ключевые слова:** раскрой; развертка полигона; геометрическое покрытие; жадный алгоритм; алгоритм «Первый подходящий»; вероятностный алгоритм.

### ВВЕДЕНИЕ

Каждое предприятие для изготовления продукции использует определенные ресурсы. От рационального их использования и минимизации отходов зависит его благополучие [1]. При изготовлении заготовок необходимо решить задачу оптимизации раскроя для сокращения количества отходов материала. Такая задача возникает в металлургии, машиностроении, лесобработывающей и лесной промышленности, а также на швейном, обувном, меховом и кожевенном производствах [2]. Нередко на заготовительном производстве приходится решать и еще одну задачу дискретной оптимизации – задачу геометрического покрытия. Как

правило, задачи геометрического покрытия являются первичными при решении более сложных технических проблем [3].

В последнее время класс задач геометрического покрытия привлекает к себе все более пристальное внимание исследователей. Причина такого любопытства заключается в их сложности и разнообразии, т. к. они относятся к классу NP-трудных задач, где при увеличении размерности возрастает их сложность [4], и для их решения не существует точных алгоритмов полиномиальной сложности. Поэтому для решения задач с малой размерностью используют точные алгоритмы [5], для задач же большой размерности годятся только эвристические подходы.

Эффективность решения задачи разбиения развертки 3D-объектов на полигональные примитивы несет существенный экономический эффект для предприятия [6], поскольку само решение представляет собой план раскроя дорогостоящего материала (например, металлических листов) на детали, из которых затем конструируется (например, при помощи сварки) трехмерная поверхность сложной формы. Например, в случае изготовления фюзеляжа самолета или подводной лодки сварочный шов подлежит дальнейшей дорогостоящей обработке (например, герметизации), поэтому критерии оптимизации и конкретизация трактовки задач могут существенно различаться от случая к случаю.

Подобные задачи рассмотрены в работах Ю. И. Валиахметовой [7] и Э. А. Мухачевой [8] и направлены на разработку алгоритма решения задачи геометрического покрытия [9]. Задача разбиения 3D-объекта на полигональные примитивы является частным случаем задач геометрического покрытия [7].

В данной статье проведен сравнительный анализ алгоритмов для трех разных объектов, цель которого – выявление эффективности и оценка быстродействия того или иного алгоритма в зависимости от выбранной фигуры.

#### ВЫБОР 3D-ФИГУР ДЛЯ ЭКСПЕРИМЕНТА

Разработка моделей 3D-поверхностей на практике представляет собой довольно сложный процесс. На начальном этапе происходит сбор информации: прорисовываются эскизы, чертежи, просматриваются видеоролики и фотографии. На основании собранных данных при помощи специализированных инструментальных средств создается трехмерная модель [10]. После построения модели объект можно рассмотреть с любого ракурса, внести коррективы.

Основной задачей исследования является анализ алгоритмов разбиения разверток поверхностей трехмерных тел на полигоны и получение минимального количества полигонов.

Очевидно, что площадь полигона зависит от развертки. Таким образом, сначала

требуется выбрать наиболее рациональную (для покрытия полигонами) развертку [11].

В данном исследовании рассматриваются плоские развертки трехмерных тел.

В качестве исследуемых выбраны три типа тел: куб, октаэдр и сфера. Но т. к. сфера не является разворачиваемой на примитивы поверхностью, вместо нее рассматриваем максимально приближенное к ней тело – курносый додекаэдр.

Для куба существует 11 различных разверток (рис. 1).

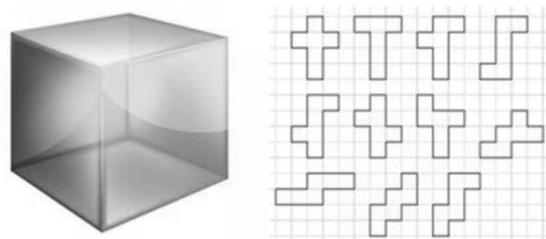


Рис. 1. Развертки куба

Чтобы выбрать оптимальную развертку, уже на данном этапе необходимо рассчитать максимальный размер полигона для каждого случая. На рис. 2 показано 11 вариантов разбиения развертки куба на отдельные геометрические примитивы с выделением одного максимального по размеру полигона, из которого возможно воспроизвести всю фигуру целиком.

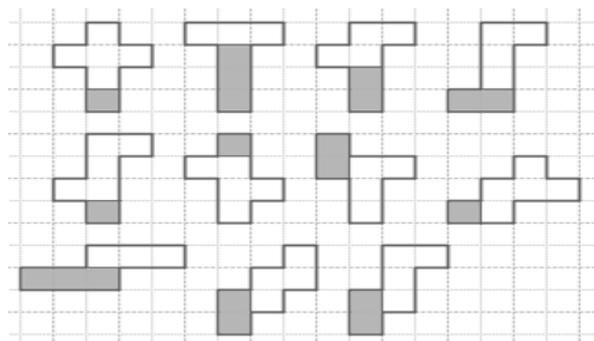


Рис. 2. Определение максимального полигона

Таким образом, наиболее подходящими являются развертки, полигоны которых представляют собой прямоугольники с соотношением сторон 3:1, поскольку количество примитивов в этом случае минимально возможное (развертка в виде буквы «Т», вторая в первом ряду на рис. 2).

На рис. 3 представлены 11 различных разверток октаэдра.

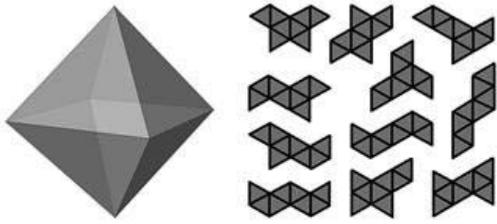


Рис. 3. Развертки октаэдра

Также октаэдр возможно представить в виде сплошной развертки, как показано на рис. 4. Типы таких разверток в данной работе не рассматриваем по причине малой площади соприкосновения отдельных полигонов между собой и, как следствие, большей площади отходов после раскроя.

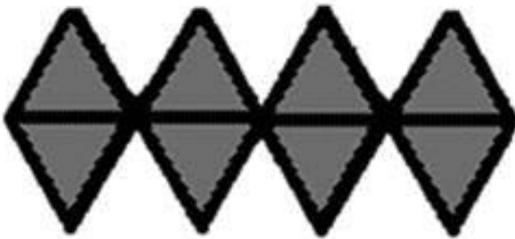


Рис. 4. Пример сплошной развертки

Используя алгоритм разбиения развертки куба на полигоны, разобьем аналогично каждую из разверток октаэдра. На рис. 5 закрашенными областями показаны максимальные полигоны, из которых возможно воспроизвести всю фигуру целиком.

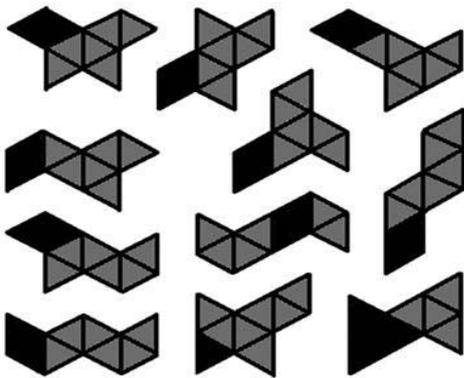


Рис. 5. Разбиение развертки на полигоны

Таким образом, наиболее подходящей является развертка, полигоны которой – равносторонние треугольники, каждая сторона которых связана с ребром октаэдра соотношением 3:1 (правая нижняя развертка на рис. 5).

На рис. 6 представлена развертка курносого додекаэдра.

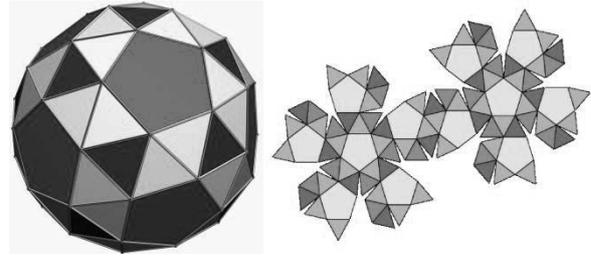


Рис. 6. Развертка курносого додекаэдра

Как видно на рис. 6, в развертке участвуют не только треугольники, но и пентагоны, которые в свою очередь можно также разбить на равные равнобедренные треугольники, т. е. в разложении курносого додекаэдра участвуют два вида примитивов.

#### ВЫБОР И РЕАЛИЗАЦИЯ АЛГОРИТМОВ ГЕОМЕТРИЧЕСКОГО ПОКРЫТИЯ РАЗВЕРТОК

Для покрытия разверток полигонами выбрали три алгоритма: «Первый подходящий», жадный и вероятностный.

*Жадный алгоритм* – это алгоритм, стремящийся сделать наиболее выгодный выбор на каждом этапе, при этом во внимание принимаются только данные текущего шага. Алгоритм не гарантирует получения оптимального решения [12–14, 20].

На вход алгоритма подается плоская развертка объекта с суммарной площадью  $S$ , заданная списком вершин  $\delta$ , полученных при обходе развертки против часовой стрелки.

На выходе необходимо получить набор примитивов, покрывающий всю площадь фигуры развертки.

*Реализация жадного алгоритма:*

1. Рассматриваются все тройки последовательных вершин входного списка  $\delta$  (развертка обходится против часовой стрелки, как показано на рис. 7), проверяется знак псевдоскалярного произведения векторов по формуле:

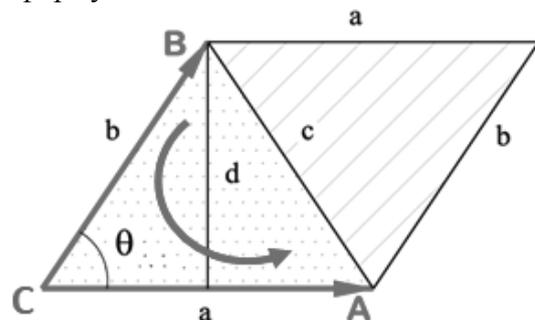


Рис. 7. Обход вершин развертки

$$a \wedge b = |a| \cdot |b| \sin \theta, \quad (1)$$

где  $\theta = \angle(a, b)$  – угол вращения (против часовой стрелки) от  $a$  к  $b$ :

– если он меньше или равен нулю, то переходят к обработке следующей тройки точек;

– если он больше нуля, то:

○ осуществляется поиск точек пересечения лучей  $CB$  и  $CA$  с ребрами развертки, из полученных точек формируется список  $\eta = \{K_j, j \in J\}$ , где  $J$  – количество найденных точек;

○ для каждой найденной точки  $K_j \in \eta$ ,  $j \in J$  осуществляется проверка принадлежности развертке треугольников  $K_jAB$ ,  $K_jAC$ ,  $K_jBC$  (рис. 8);

○ для каждой пары найденных точек  $K_j, K_i \in \eta$ ,  $i, j \in J$  осуществляется проверка принадлежности развертке треугольника  $K_iK_jC$ ;

○ для каждой найденной точки  $K_j \in \eta$ ,  $j \in J$ , лежащей на луче  $CB$ , и каждой точки  $P \in \delta$ , не лежащей на лучах  $CB$  и  $CA$ , осуществляется проверка принадлежности развертке четырехугольника  $CK_jPA$ ;

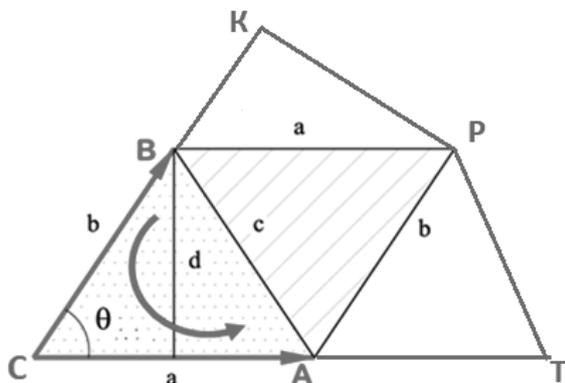


Рис. 8. Пример расположения точек  $K$  и  $T$

○ для каждой найденной точки  $K_i \in \eta$ ,  $i \in J$ , лежащей на луче  $CA$ , и каждой точки  $P \in \delta$ , не лежащей на лучах  $CB$  и  $CA$ , осуществляется проверка принадлежности развертке четырехугольника  $CBPK_i$ ;

– при выявлении принадлежности соответствующий треугольник или четырехугольник включается в список  $\tau$  примитивов, покрывающих развертку.

2. Список полигонов покрытия  $\tau = \{\tau_i, i \in N\}$ , где  $N$  – общее количество сгенерированных полигонов, сортируется по невозрастанию площадей полигонов, т. е.

каждый следующий элемент списка не превосходит предыдущий  $\forall i \in N: \tau_i \geq \tau_{i+1}$ .

3. Осуществляется последовательный проход по списку  $\tau$ , выделяется минимальное множество непересекающихся полигонов, образующих точное геометрическое покрытие исходной области развертки.

«Первый подходящий» алгоритм является простой однопроходной эвристикой, с помощью которой на каждом шаге работы алгоритма для добавления к геометрическому покрытию развертки выбирается первый допустимый примитив [15]. Покрытие происходит в направлении слева направо, сверху вниз. Алгоритм лишь дает приближительное решение и не гарантирует получение хорошего результата [16].

Реализация алгоритма «Первый подходящий»:

1. Из списка  $\delta$  выбираются случайным образом 3 вершины, не обязательно последовательные, такие, что данная тройка на предыдущих этапах алгоритма не рассматривалась.

2. Точки сортируются по возрастанию номеров в списке  $\delta$ . Пусть после этого мы имеем тройку точек  $A, B, C$ .

3. Проверяется знак псевдоскалярного произведения векторов  $CB \wedge CA$ :

– если  $CB \wedge CA \leq 0$ , то этот набор точек отбрасывается, возвращаемся к п. 1;

– иначе проверяется, принадлежит ли вся площадь треугольника  $ABC$  области развертки. При выявлении принадлежности область треугольника  $ABC$  исключается из исходной области развертки, и примитив  $ABC$  добавляется к списку  $\tau$ , и корректируется список  $\delta$  путем исключения вершин  $A, B, C$  и возможного включения образовавшихся новых вершин. При этом исходная область развертки может оказаться разбитой на изолированные части, для каждой из которых составляется список обхода вершин  $\delta_i$ , где  $i$  – порядковый номер рассматриваемой изолированной части, и рекурсивно применяется алгоритм, начиная с п. 1.

Замечание: целесообразно ограничить сверху количество итераций обработки каждой изолированной области и исключать из рассмотрения область по их истечении. В данном алгоритме после 100 неудачных

попыток сгенерировать полигон рассматриваемая область исключается из рассмотрения.

В *вероятностном алгоритме* каждому найденному полигону, входящему в исходную развертку, ставится в соответствие вероятность использования, пропорциональная его площади. Работа происходит со списком полигонов, упорядоченных по невозрастанию вероятности использования [18, 21], т. е. каждый следующий элемент списка не превосходит предыдущий.

Далее покрытие исходной области развертки происходит заданное количество раз по принципу алгоритма «Первый подходящий». В алгоритме используется доверительный коэффициент, начальное значение которого равно единице. Единственный управляемый коэффициент – это детализация доверительности [19]. Он показывает количество шагов одинаковой величины, за которые доверительный коэффициент уменьшится до нуля, что определяет количество доверительных циклов [8].

*Реализации вероятностного алгоритма:*

1. На вход вероятностного алгоритма поступает множество  $\Omega = \{\sigma_i\}$ ,  $i = 1...N$  готовых наборов полигонов, каждый из которых представляет собой покрытие исходной области развертки, полученное методом «Первый подходящий». Формируется список  $\gamma$  всех типов полигонов, входящих в  $\Omega$ , с исключением повторяющихся элементов сформированного списка. Список  $\gamma$  упорядочивается по невозрастанию частоты появления полигона в  $\Omega$ , при этом возможны

случаи, когда у разных элементов частота появления одинакова.

2. Последовательно просматриваются полигоны из списка  $\gamma$ . Для каждого такого полигона происходит попытка расположения его в площади развертки (в различных точках развертки и при различных углах поворота полигона).

– Если удалось разместить полигон, то его положение закрепляется, и он заносится в итоговый список  $\tau$ .

– Для каждой замкнутой изолированной области развертки алгоритм повторяется со 2-го шага.

*Замечание:* как и в предыдущем пункте, если на некотором участке не удастся сгенерировать полигон, то после определенного количества итераций область исключается из рассмотрения. Авторский алгоритм после 1000 неудачных попыток генерации полигона исключает область из рассмотрения.

## ПРОВЕДЕНИЕ ЭКСПЕРИМЕНТА

Для проведения эксперимента была создана программа. На главном окне программы (рис. 9) находятся три активные кнопки выбора 3D-моделей. При выборе одной из моделей становится активной кнопка ввода параметров для создания 3D-объекта, а также в правой части экрана появляется визуализация выбранной модели. Кнопка «создать развертку» становится активной только после того, как будет выполнено построение 3D-объекта.

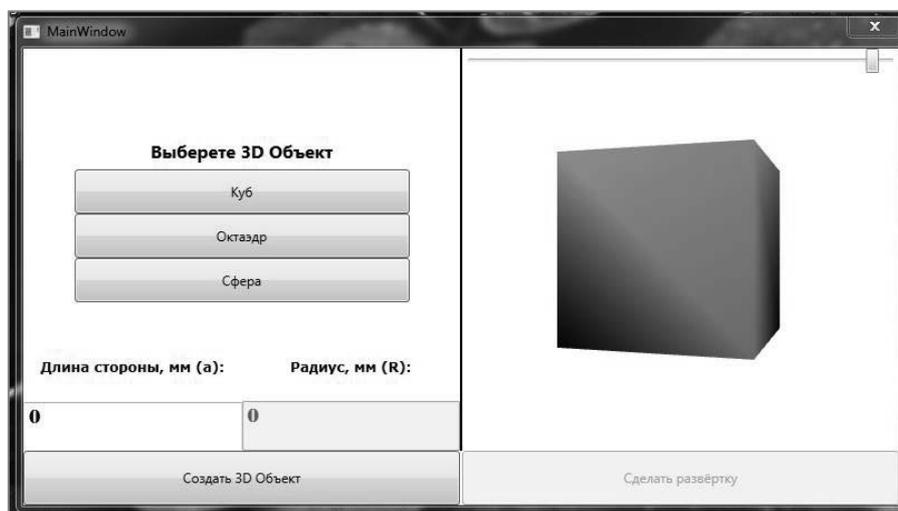


Рис. 9. Главное окно программы

На рис. 10 показана развертка выбранного объекта с заданными параметрами, в данном случае развертка куба. Размер развертки автоматически масштабируется при отрисовке на форме. Предусмотрена возможность проведения расчетов тремя описанными выше алгоритмами, которые запускаются при нажатии на соответствующую кнопку.

После выбора алгоритма появляются две формы: форма с данными о полученном по-

крытии (рис. 11) и форма с геометрическим покрытием развертки (рис. 12).

Для тестирования программы в качестве 3D-объекта был выбран октаэдр.

В поле ввода длин сторон было введено значение 40, единица измерения – миллиметры (рис. 13).

Для обработки развертки октаэдра были проведены расчеты жадным алгоритмом, алгоритмом «Первый подходящий» и вероятностным алгоритмом. На рис. 14–16 представлены результаты работы.

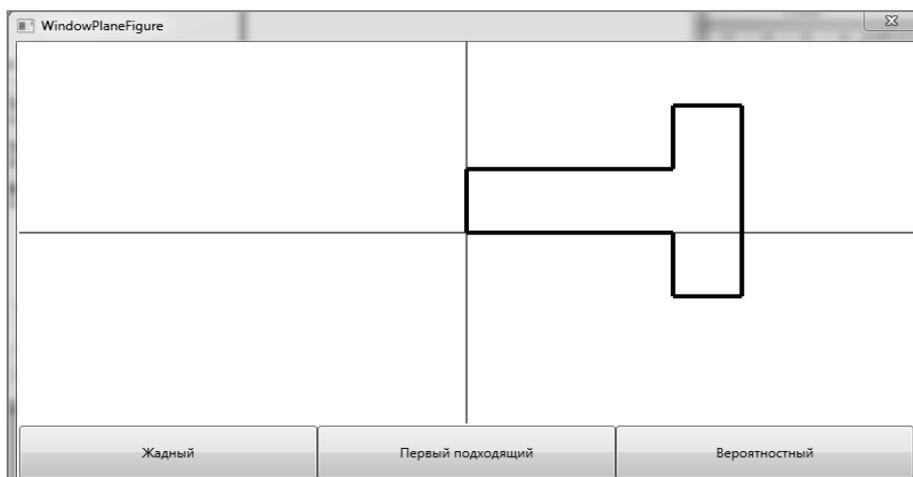


Рис. 10. Развертка куба

Таблица Полигонов					Информация
Id	Point1	Point2	Point3	S, мм	<b>Алгоритм полигонизации:</b>  ЖАДНЫЙ
0	30,-20	40,-20	30,0	3600	
1	40,-20	40,10	30,10	5400	
2	30,10	30,0	40,-20	1800	
3	30,0	0,0	0,-10	5400	
4	0,-10	30,-10	30,0	5400	<b>Время выполнения алгоритма:</b>  00:00:00.0030002
					<b>Количество полигонов:</b>  5 Polys ( Покрыто площади: 100% )

Рис. 11. Данные о полученном геометрическом покрытии

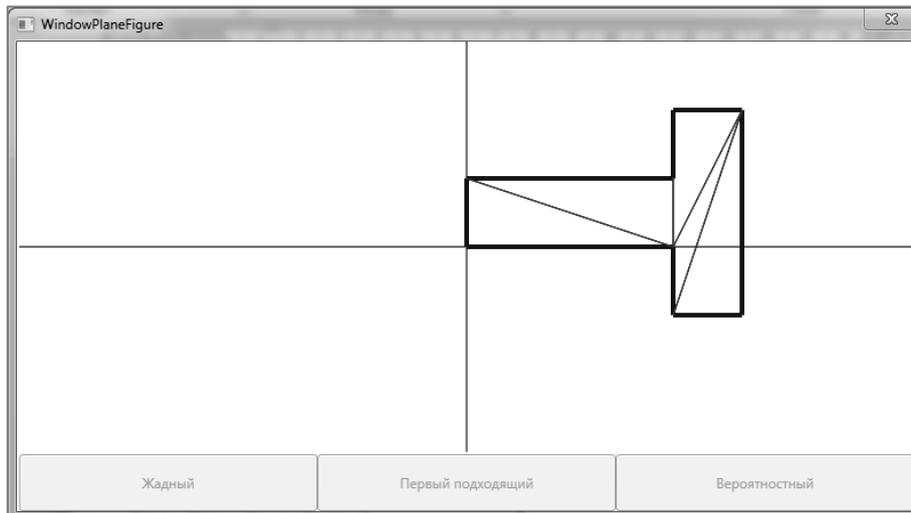


Рис. 12. Геометрическое покрытие треугольными примитивами развертки куба

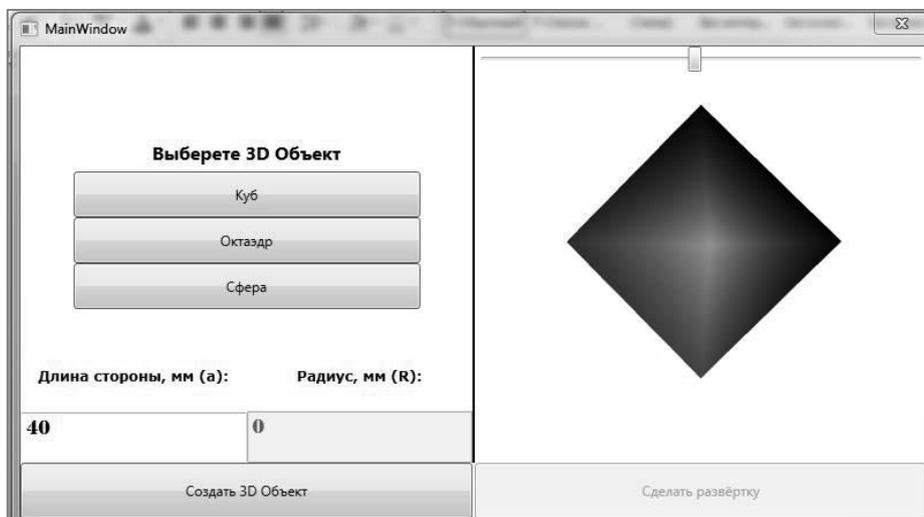


Рис. 13. Ввод параметров для октаэдра

Таблица Полигонов					Информация	
Id	Point1	Point2	Point3	S, мм	<b>Алгоритм полигонизации:</b>	
0	0.0	40.0	20.34	2771.28	ЖАДНЫЙ	
1	30.17	50.51	10.51	2771.28	<b>Время выполнения алгоритма:</b>	
					00:00:00.0119953	
					<b>Количество полигонов:</b>	
					2 Polys ( Покрыто площади: 99% )	

Рис. 14. Результат работы жадного алгоритма (для обработки развертки октаэдра)



Рис. 15. Результат работы алгоритма «Первый подходящий» (для обработки развертки октаэдра)



Рис. 16. Результат работы вероятностного алгоритма (для обработки развертки октаэдра)

Чтобы сделать выводы о работоспособности алгоритмов покрытия провели эксперименты, результаты некоторых из них приведены в табл. 1–3.

Для обработки развертки курносого додекаэдра были проведены расчеты аналогичными алгоритмами. На рис. 17–19 представлены результаты работы.

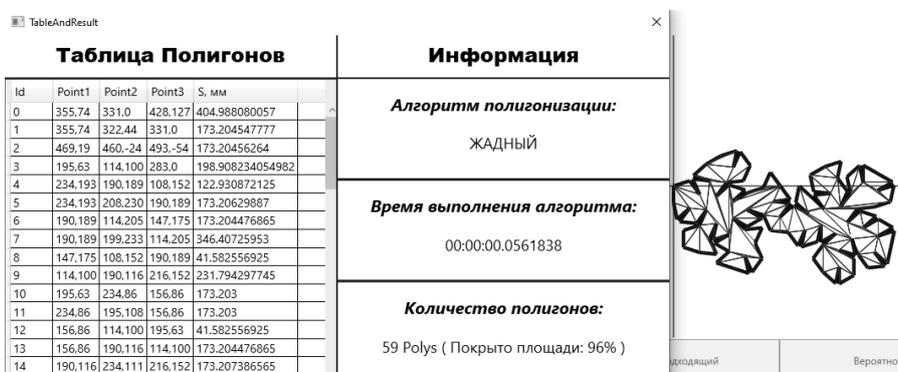


Рис. 17. Результат работы жадного алгоритма (для обработки развертки курносого додекаэдра)

Таблица Полигонов					Информация	
Id	Point1	Point2	Point3	S, мм		
0	575.172	534.209	482.192	173.208461	<b>Алгоритм полигонизации:</b>	
1	575.172	587.226	534.209	173.2010266	"ПЕРВЫЙ ПОДХОДЯЩИЙ"	
2	582.291	534.319	534.264	173.201732	<b>Время выполнения алгоритма:</b>	
3	534.264	523.318	450.236	321.8433032	00:00:00.8708609	
4	523.318	482.281	450.236	41.5845738	<b>Количество полигонов:</b>	
5	450.236	582.236	534.264	240.485	57 Polys ( Покрыто площади: 96% )	
6	582.236	582.291	534.264	173.201732		
7	19.116	84.27	116.-16	0.000360262099974307		
8	0.55	0.0	84.27	307.76800000001		
9	0.0	52.-16	84.27	190.211044056		
10	52.71	11.108	0.55	173.205335215513		
11	0.55	84.27	52.71	190.210428520016		
12	84.27	62.-22	116.-16	173.2047742425		
13	346.100	387.63	423.144	307.7750164		
14	116.172	116.-16	139.33	280.254191196		

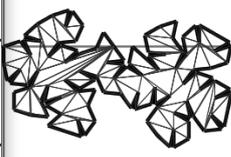


Рис. 18. Результат работы алгоритма «Первый подходящий» (для обработки развертки курносого додекаэдра)

Таблица Полигонов					Информация	
Id	Point1	Point2	Point3	S, мм		
0	237.49	295.37	259.69	321.83036765	<b>Алгоритм полигонизации:</b>	
1	248.9	296.-43	290.-14	240.478819845	ВЕРоятностный	
2	248.9	267.-34	296.-43	240.483171945	<b>Время выполнения алгоритма:</b>	
3	248.9	241.-19	267.-34	190.215274895	00:00:00.7761485	
4	290.-14	277.13	248.9	190.215683595	<b>Количество полигонов:</b>	
5	295.37	265.40	248.16	173.2026248	68 Polys ( Покрыто площади: 96% )	
6	214.29	221.0	265.40	346.409216136		
7	265.40	295.37	237.49	41.5804322		
8	237.49	214.29	265.40	173.20411675		
9	248.16	277.13	295.37	173.20872032		
10	277.13	290.-14	307.9	173.20185176		
11	307.9	295.37	277.13	173.200806475		
12	130.42	104.57	46.69	122.929938275		
13	130.42	156.57	104.57	173.203		
14	156.57	130.72	104.57	173.203		

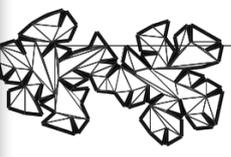


Рис. 19. Результат работы вероятностного алгоритма (для обработки развертки додекаэдра)

По полученным результатам можно сделать вывод, что жадный и «Первый подходящий» алгоритмы для покрытия развертки куба и октаэдра использовали минимальное количество полигонов, в отличие от вероятностного алгоритма.

Вне зависимости от количества полигонов, входящих в покрытие, все три алгоритма полностью покрыли развертку. Время

выполнения алгоритмов сравнимо, что объясняется небольшой размерностью рассматриваемых задач.

Результаты по покрытию разверток куба, октаэдра и курносого додекаэдра приведены в табл. 1–3 соответственно, где процент заполнения рассчитывается по формуле с отбросом дробной части.

Таблица 1

#### Результаты покрытия развертки куба

Сторона	Жадный алгоритм			«Первый подходящий» алгоритм			Вероятностный алгоритм		
	Время вып., с	Кол-во полигонов, шт.	% заполнения	Время вып., с	Кол-во полигонов, шт.	% заполнения	Время вып., с	Кол-во полигонов, шт.	% заполнения
10	0,0132153	2	100	0,0009958	2	100	0,0160756	3	100
20	0,0009972	2	100	0,0009968	2	100	0,0141142	4	100
30	0,0009970	2	100	0,0009670	2	100	0,0114133	6	100
40	0,0009973	2	100	0,0009971	2	100	0,0147054	4	100
50	0,0009940	2	100	0,0009969	2	100	0,0145108	4	100

Таблица 2

## Результаты покрытия развертки октаэдра

Сторона	Жадный алгоритм			«Первый подходящий» алгоритм			Вероятностный алгоритм		
	Время вып., с	Кол-во полигонов, шт.	% заполнения	Время вып., с	Кол-во полигонов, шт.	% заполнения	Время вып., с	Кол-во полигонов, шт.	% заполнения
10	0,0132153	2	99	0,0109819	2	99	0,0132075	2	99
20	0,0141414	2	99	0,0140095	2	99	0,0138582	3	99
30	0,0138740	2	99	0,0141304	2	99	0,0105763	3	99
40	0,0103668	2	99	0,0089884	2	99	0,0141811	3	99
50	0,0139250	2	99	0,0138252	2	99	0,0159913	2	99

Таблица 3

## Результаты покрытия развертки курносого додекаэдра

Радиус	Жадный алгоритм			«Первый подходящий» алгоритм			Вероятностный алгоритм		
	Время вып., с	Кол-во полигонов, шт.	% заполнения	Время вып., с	Кол-во полигонов, шт.	% заполнения	Время вып., с	Кол-во полигонов, шт.	% заполнения
10	0,0505647	40	96	0,6292075	58	96	0,8881405	57	96
20	0,8769631	57	96	0,8748084	48	96	0,8155128	90	96
30	0,8742424	55	96	0,0319185	59	96	0,0569684	68	96
40	0,0561838	59	96	0,0694573	47	96	0,7761485	68	96
50	0,8722528	52	96	0,8708609	57	96	0,8820773	96	96

## ЗАКЛЮЧЕНИЕ И ВЫВОДЫ

Проблема экономии ресурсов остается актуальной на многих производственных предприятиях. Задачи геометрического покрытия 3D-объектов возникают в ресурсоемких направлениях, и, как правило, автоматизация решения подобных задач влечет за собой серьезный экономический эффект.

В ходе исследования разработаны и реализованы алгоритмы геометрического покрытия развертки трехмерной поверхности. Проведен эксперимент, в ходе которого рассмотрены развертки для выбранных 3D-моделей.

Результаты эксперимента показали, что для покрытия развертки куба, октаэдра и курносого додекаэдра целесообразно использовать жадный или «Первый подходящий» алгоритмы.

В планы авторов входит дальнейшее исследование и усовершенствование разработанных алгоритмов и разработка новых. При этом для задач большой размерности планируется использовать эвристические подходы.

Представляет большой интерес обработка разверток сложных трехмерных объектов, таких как элементы фюзеляжа самолета, кузова автомобиля и др.

## СПИСОК ЛИТЕРАТУРЫ

1. **Жданова Л. А.** Организация и управление промышленной фирмой в развитых странах: учебник. М.: Экономика, 2010. 263 с. [ L. A. Zhdanova, *Organization and management of an industrial firm in developed countries: textbook*, (in Russian). Moscow: Ekonomika, 2010. ]
2. **Мухачева А. С., Сурначев М. Ю.** Задача параллелепипедной упаковки: декодер на базе блочных структур // Принятие решений в условиях неопределенности: межвуз. науч. сб. Уфа: УГАТУ, 2005. С. 51–55. [ A. S. Mukhacheva, M. Yu. Surnachev, "Parallelepiped packing problem: decoder based on block structures", (in Russian), in *int. scientific collection of works Decision-Making in conditions of uncertainty*. Ufa: UGATU, 2005, pp. 51-55. ]
3. **Кузнецов В. Ю.** Методы покрытия многосвязных ортогональных многоугольников для задач оптимального размещения сенсоров в области мониторинга: дис. ... канд. техн. наук: 05.13.18. Уфа, 2009. 162 с.: ил. [ V. Yu. Kuznetsov, *Methods of covering multiply connected orthogonal polygons for problems of optimal placement of sensors in the field of monitoring: PhD*, (in Russian). Ufa, 2009. ]
4. **Гэри М., Джонсон Д.** Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982. 466 с. [ M. Gary, D. Johnson, *Computing Machines and Difficult Problems*, (in Russian). Moscow: Mir, 1982. ]
5. **Baker B. S.** Approximation Algorithms for NP-Complete Problems on Planar Graphs // *Journal of the ACM*. 1994. No. 41 (1). Pp. 153-180. [ B. S. Baker, "Approximation Algorithms for NP-Complete Problems on Planar Graphs", in *Journal of the ACM*, no. 41 (1), pp. 153-180, 1994. ]
6. **Minimum** edge length partitioning of rectilinear polygons / A. Lingas, et al. // *Proc. 20th Allerton Conf. Comput.*, 1982. Pp. 53-63. [ A. Lingas, et al., "Minimum edge length partitioning of rectilinear polygons", in *Proc. 20th Allerton Conf. Comput.*, 1982, pp. 53-63. ]

7. **Валиахметова Ю. И., Горбатов В. А., Пятаев Н. М.** Разбиение многосвязного ортогонального полигона с минимизацией протяженности стыков на основе линейного программирования // Молодой ученый. 2019. № 20 (258). С. 1–5. [ Yu. I. Valiakhmetova, V. A. Gorbatov, N. M. Pyataev, "Partitioning of a multiply connected orthogonal polygon with minimization of the length of joints on the basis of linear programming", (in Russian), in *Molodoy uchenyj*, no. 20 (258), pp. 1-5, 2019. ]

8. **Мухачева Э. А.** Рациональный раскрой промышленных материалов. Применение АСУ. М.: Машиностроение, 1984. 176 с.: ил. [ E. A. Mukhacheva, *Rational cutting of industrial materials. Application of ACS*, (in Russian). Moscow: Mashinostroenie, 1984. ]

9. **NuMVC: An Efficient Local Search Algorithm for Minimum Vertex Cover / Shaowei Cai, et al.** // J. Artif. Int. Res. 2013. Vol. 46, no. 1. Pp. 687-716. [ Shaowei Cai, et al., "NuMVC: An Efficient Local Search Algorithm for Minimum Vertex Cover", in *J. Artif. Int. Res.*, vol. 46, no. 1, pp. 687-716, 2013. ]

10. **Гайсина С. В., Огановская Е. Ю., Князева И. В.** Робототехника, 3D-моделирование, прототипирование в дополнительном образовании. Реализация современных направлений. СПб.: Каро, 2017. 209 с. [ S. V. Gaisina, E. Yu. Oganovskaya, I. V. Knyazeva, *Robotics, 3D modeling, prototyping in additional education. Implementation of modern trends*, (in Russian). St. Petersburg: Karo, 2017. ]

11. **Dahl O.-J., Hoare C. A. R.** Hierarchical Program Construction in Structured Programming. New York: Academic-Press, 1972. Pp. 174-220. [ O.-J. Dahl, C. A. R. Hoare, *Hierarchical Program Construction in Structured Programming*. New York: Academic-Press, 1972. ]

12. **Gonzalez T. F. (ed.)** Handbook of Approximation Algorithms and Metaheuristics. Taylor & Francis LLC, 2007. Pp. 802-815. [ T. F. Gonzalez (ed.), *Handbook of Approximation Algorithms and Metaheuristics*. Taylor & Francis LLC, 2007. ]

13. **Алгоритмы: построение и анализ / Т. Кормен [и др.]; пер. с англ. 2-е изд. М.: Вильямс, 2005.** [ T. Cormen, et al., *Algorithms: construction and analysis, 2nd edition*, (in Russian). Moscow: Vilyams, 2005. ]

14. **Roughgarden T.** Algorithms Illuminated (Part 3): Greedy Algorithms and Dynamic Programming. Soundlikeyourself Publishing, LLC, 2019. 230 p. [ T. Roughgarden, *Algorithms Illuminated (Part 3): Greedy Algorithms and Dynamic Programming*. Soundlikeyourself Publishing, LLC, 2019. ]

15. **Beasley J. E.** Bounds for Two-Dimensional Cutting // The Journal of the Operation Research Society. 1985. Vol. 36, No. 1. Pp. 71-74. [ J. E. Beasley, "Bounds for Two-Dimensional Cutting", in *The Journal of the Operation Research Society*, vol. 36, no. 1, pp. 71-74, 1985. ]

16. **Sedgwick R.** Algorithms. Addison-Wesley Publishing, 1983. [ R. Sedgwick, *Algorithms*, Addison-Wesley, 1983. ]

17. **Wladston F. F.** Learn the Art of Solving Computational Problems. New Jersey: Code Energy LLC, 2017. 180 p. [ F. F. Wladston, *Learn the Art of Solving Computational Problems*. New Jersey: Code Energy LLC, 2017. ]

18. **Телицкий С. В., Филиппова А. С.** Комплексный подход к решению задачи покрытия области заготовками неопределенных размеров // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2012. № 2 (145). С. 61–67. [ S. V. Telitskiy, A. S. Filippova,

"An integrated approach to solving the problem of covering an area with blanks of indefinite dimensions", (in Russian), in *Nauchno-tekhicheskie vedomosti SPbGPU. Informatika. Telekommunikacii. Upravlenie*, no. 2 (145), pp. 61-67, 2012. ]

19. **Dorigo M.** Optimization, Learning and Natural Algorithms: PhD thesis. Italy: Politecnico di Milano, 1992. [ M. Dorigo, *Optimization, Learning and Natural Algorithms*. PhD thesis. Italy: Politecnico di Milano, 1992. ]

20. **Филиппова А. С., Валиахметова Ю. И., Дямина Э. И.** Конструирование алгоритмов геометрического размещения на основе матричной и уровневой технологий // Вестник Уфимского государственного авиационного технического университета. 2016. Т. 20, № 4 (74). С. 114–124. [ A. S. Filippova, Yu. I. Valiakhmetova, E. I. Dyaminova, "Design of geometric placement algorithms based on matrix and level technologies", (in Russian), in *Vestnik UGATU*, vol. 20, no. 4 (74), pp. 114-124, 2016. ]

21. **Филиппова А. С., Дямина Э. И., Валиахметова Ю. И.** Метод ограниченной декомпозиции для решения комплексной задачи геометрического покрытия и раскроя // Информационные технологии. 2016. Т. 22, № 3. С. 179–187. [ A. S. Filippova, E. I. Dyaminova, Yu. I. Valiakhmetova, "The method of limited decomposition for solving the complex problem of geometric covering and cutting", (in Russian), in *Informacionnye tehnologii*, vol. 22, no. 3, pp. 179-187, 2016. ]

#### ОБ АВТОРАХ

**ЗЫКИН Сергей Анатольевич**, ст. преп. каф. ОНД (ЛФ ПНИПУ). Дипл. инж. АСОиУ (ПГТУ, 2011). Готовит дис. о методах и алгоритмах покрытия материалов.

**ПЕТРЕНКО Александр Анатольевич**, доц. каф. ИТАС (ПНИПУ). Дипл. «Прикладная математика» (КБГУ, 1996). Канд. техн. наук (ИжГТУ, 2013). Иссл. в обл. моделирования процессов.

**ВАЛИАХМЕТОВА Юлия Ильясовна**, доц. каф. ВМиК (УГАТУ). Дипл. инж.-прогр. (УГАТУ, 2004). Канд. техн. наук (УГАТУ, 2008). Иссл. в обл. оптимизац. задач размещения и маршрутизации.

#### METADATA

**Title:** Splitting into polygonal primitives sweep a 3D object's.

**Authors:** S. A. Zykin<sup>1</sup>, A. A. Petrenko<sup>2</sup>, J. I. Valiakhmetova<sup>3</sup>

**Affiliation:**

<sup>1</sup> Lysva branch Perm National Research Polytechnic University (LB PNRPU), Russia.

<sup>2</sup> Perm National Research Polytechnic University (PNRPU), Russia.

<sup>3</sup> Ufa State Aviation Technical University (UGATU), Russia.

**Email:** <sup>1</sup>zykin\_sergey@mail.ru, <sup>2</sup>hawk321@mail.ru, <sup>3</sup>julikas@inbox.ru

**Language:** Russian.

**Source:** Vestnik UGATU (scientific journal of Ufa State Aviation Technical University), vol. 24, no. 4 (90), pp. 101-112, 2020. ISSN 2225-2789 (Online), ISSN 1992-6502 (Print).

**Abstract:** The problem of saving resources in production remains relevant for many areas of human activity. When making blanks, it is necessary to solve the problem of optimal cutting. To solve this problem, one of the most profita-

ble methods of cutting material is selected to reduce the amount of waste when cutting blanks. Geometric coverage problems are often one of the stages in solving larger discrete optimization problems. It is necessary to position certain objects on the surface of another, usually larger object so that the area of the object to be covered is completely filled with minimization of overlapping of the covering objects. The results obtained allow us to see the rational coverage map obtained by three algorithms. The practical significance of the work is due to a significant increase in the role of rational coating of the material in connection with the expediency of minimizing waste in production.

**Key words:** cutting; polygon unfolding; geometric coverage; greedy algorithm; first fit algorithm; probabilistic algorithm.

**About authors:**

**ZYKIN, Sergey Anatolievich**, Senior lecturer, Dept. of General scientific disciplines (LB PNRPU). Dipl. engineer automated processing and control systems (PSTU, 2011).

**PETRENKO, Alexandr Anatolievich**, Assoc. Prof., Dept. of Information technologies and automated systems (PNRPU). Dipl. applied mathematics (KBSU, 1996). Cand. of Tech. Sci. in system analysis and modeling. (IzhSTU, 2013).

**VALIAKHMETOVA, Yulia Il'yasovna**, Assoc. Prof., Dept. of Computational mathematics and Cybernetics (UGATU). Dipl. software engineer (UGATU, 2004). Cand. of Tech. Sci. (UGATU, 2008).