

М. А. Волохов, А. В. Дроботов

**МЕТОДЫ ПРЕОБРАЗОВАНИЯ И РАЗДЕЛЕНИЯ 3D-МОДЕЛИ
НА СЛОИ ДЛЯ АДДИТИВНОГО ПРОИЗВОДСТВА**

Волгоградский государственный технический университет

E-mail: maxim_volohov@mail.ru

Рассматривается проблема сложности и длительности подготовки моделей к 3D-печати, с целью упрощения и ускорения этого трудоемкого процесса. Описана микросервисная система, которая генерирует G-код по запросу пользователя, особенности работы с форматами 3D-моделей и их преобразованием.

Ключевые слова: 3D-принтер, автоматизация, 3D-печать, обработка информации, слайсинг, преобразование моделей, микросервисная архитектура, 3MF-формат.

M. A. Volokhov, A. V. Drobotov

METHODS FOR CONVERTING AND DIVIDING A 3D-MODEL INTO LAYERS

Volgograd State Technical University

The problem of the complexity and duration of preparing models for 3D-printing is considered in order to simplify and accelerate this time-consuming process. A micro-service system that generates a G-code at the user's request, features of working with 3D-model formats and their transformation are described.

Keywords: 3D-printer, automation, 3D-printing, information processing, slicing, model transformation, micro-service architecture, 3MF-format.

Аддитивное производство произвело революцию в промышленности, позволив создавать изделия сложной геометрии при низком расходе материалов, зачастую выигрывая также в себестоимости у традиционных методов изготовления. Однако одним из ограничений аддитивного производства является сложность и длительность подготовки 3D-моделей к печати, так как оператор проводит ряд действий вручную. Одним из этапов подготовки к печати является нарезка (деление) модели на слои и генерация

G-кода – алгоритма производства изделия на машинном языке. В этой работе предлагается система упрощенной и ускоренной подготовки 3D-моделей к печати в облачном сервисе и описаны основные особенности взаимодействия разрабатываемой системы с программами нарезки 3D-модели на слои.

Разрабатываемый облачный сервер состоит из двух микросервисов: *rabbitmq-server* и *slicer-engine-server*, изображенных на рис. 1.



Рис. 1. Схема организации разрабатываемого облачного сервера

Первый микросервис в облачном сервисе – *rabbitmq-server* – является промежуточным звеном между клиентом и микросервисом нарезки модели *slicer-engine-server*. Он предварительно обрабатывает нарезаемую модель и формирует сообщение во второй сервис. Микросервис *rabbitmq-server* подготовки содержит в основе себя брокер сообщений *RabbitMQ* на основе протокола *AMQP*, брокер позволяет общаться сервису с клиентской частью системы, от которой поступает запрос на подготовку модели. Сервер *RabbitMQ* имеет обменник, который складывает сообщения в две очереди: одна – для получения сообщений запроса подготовки от клиента, а другая – для отправки итогового G-кода обратно клиенту. В запросе передаются данные в *JSON* формате в байтовом представлении. Они включают уникальный номер запроса, ссылку на 3D-модель, параметры ис-

пользуемых экструдеров, а также глобальные настройки печати.

Когда сообщение запроса от клиента попадает в очередь, на которую есть подписка у *rabbitmq-server*, то микросервис считывает его и удаляет из очереди. Затем идет чтение запроса, в начале которого определяется формат 3D-модели, от которого зависит алгоритм последующей обработки. Если модель имеет формат *STL* (набор треугольных полигонов, образующих поверхность 3D-модели), то из нее определяются массивы вершин и индексов, формирующих полигоны. В одном полигоне находится три вершины, однако вершина может быть общей для нескольких полигонов. На рис. 2 видно, что индексы состоят из трех чисел, указывающих на номера вершин полигона. Так индексы [3, 4, 5] и [4, 3, 6] будут иметь две общие вершины 3 и 4, а значит – общую сторону.

Формат STL	Формат 3MF
Вершины модели	Вершины модели
[[-1.0000000e+01 2.0000000e+01 1.0000000e+01	[[-10. 10. 10.]
1.0000000e+01 2.0000000e+01 1.0000000e+01]	[10. 10. 10.]
[-1.0000000e+01 2.0000000e+01 -1.0000000e+01]	[-10. 10. -10.]
1.0000000e+01 4.04357911e-07 -1.0000000e+01]	[10. -10. -10.]
[-1.0000000e+01 4.04357911e-07 1.0000000e+01]	...
[-1.0000000e+01 4.04357911e-07 -1.0000000e+01]	[-10. -10. 10.]
1.0000000e+01 4.04357911e-07 1.0000000e+01]	[10. 10. -10.]
1.0000000e+01 2.0000000e+01 -1.0000000e+01]]	[10. -10. -10.]
Индексы вершин (формируют полигоны)	Индексы вершин (формируют полигоны)
[[0 1 2]	[[0 1 2]
[3 4 5]	[3 4 5]
[4 3 6]	[6 7 8]
[7 2 1]	[9 10 11]
[4 6 1]	[12 13 14]
[4 2 5]	[15 16 17]
[7 1 6]	[18 19 20]
[5 2 7]	[21 22 23]
[4 0 2]	[24 25 26]
[6 3 7]	[27 28 29]
[1 0 4]	[30 31 32]
[7 3 5]]	[33 34 35]]

Рис. 2. Отличие полигонов между STL и 3MF-форматом

Модель на обработку может поступить в формате 3MF, который был разработан на основе XML формата специально для 3D-печати. Он представляет из себя архив проекта, содержащий файл с моделью и параметры проекта. У модели в формате 3MF индексы вершин идут по порядку, потому что в массиве вершин модели указываются координаты всех вершин и для каждого полигона уже в нужном порядке. Можно заметить, что в массиве вершин STL-формата нет вершин с одинаковыми координатами, а в формате 3MF они присутствуют.

Важным преимуществом 3MF-формата является содержание дополнительной информации, например, единицы измерения, данных по цветам, поддержкам, а также положению в пространстве относительно рабочей области печати. Положение модели задается относительно нижне-левого угла в виде матрицы перемещения. С помощью матрицы преобразований помимо перемещения также задается масштабирование и поворот модели относительно каждой оси. На рис. 3 отображены смещения по осям X, Y и Z в последнем столбце, если прибавить эти координаты к каждой вершине модели, то она сместится на центр рабочего стола.

Трехмерная матрица преобразования			
[[	1.	0.	157.5
[	0.	0.	150.1499939]
[	0.	1.	10.
[	0.	0.	1.

Рис. 3. Трехмерная матрица преобразования 3MF-модели

Важной особенностью работы с 3D-моделями является определение направлений осей Y

и Z координатной системы. Это связано с тем, что в разных программах и системах используется разная координатная сетка. Для решения этой проблемы необходимо повернуть систему координат, поменяв оси Z и Y местами, а также умножив все координаты вершин по оси Y на -1.

3MF также поддерживает наличие групп и составных моделей. Если в файле несколько изделий, то матрица преобразования будет составляться для каждой модели отдельно.

Второй микросервис в облачном сервисе – *slicer-engine-server* – отвечает за нарезку модели и генерацию G-кода. Для разделения модели на слои используется открытое консольное приложение *CuraEngine*, но для него данные сначала нужно подготовить в формате сообщений *ArcusMessage*, чтобы передавать между сокетами. По умолчанию *CuraEngine* может нарезать модель только на плоские горизонтальные слои, однако изменяя алгоритмы движка, можно добиться и других форм нарезки. После расшифровки массива глобальных настроек печати, они приводятся к более удобному и читаемому виду, когда тип параметра исходит из его названия. Если в пришедшем сообщении установлен режим печати "classic", то используется *CuraEngine*, для которого объявляются типы сообщений, устанавливается IP адрес и номер порта для взаимодействия.

Если во время считывания сообщения модели от *RabbitMQ* микросервиса данные матрицы преобразования пусты, то находится нижняя координата модели, потому что модель может находиться ниже рабочей области. На рис. 4 слева отображена модель с пустой матрицей

преобразования, а справа модель, перенесенная на центр и поднятая из-под рабочей области печати.

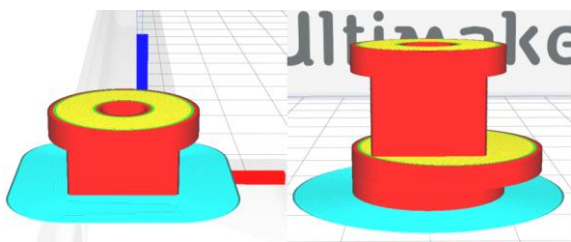


Рис. 4. Модель до и после преобразования

Для определения нижней точки 3D-модели создается сетка по присланным вершинам и индексами вычисляется аппроксимация выпуклой оболочки массива вершин, которая возвращает минимальные и максимальные координаты модели для каждой оси. Этих значений хватит для определения ширины, высоты, глубины, центра и каждой стороны нарезаемой модели. Также для модели создается новая матрица преобразования с масштабом 1:1:1 и перемещением на центр рабочего стола, ширина и глубина области печати берутся из прочитанных глобальных настроек. Все координаты по оси Y поднимаются на величину, которую имеет нижняя координата модели, это необходимо для поднятия фигуры на рабочий стол.

Из полученной матрицы преобразования берется матрица масштабирования и матрица перемещения, изображенные на рис. 5, а, б. Первая перемножается на вершины модели, а вторая суммируется с координатами A, B и C. После данного преобразования производится смена осей Y и Z. На этом этапе модель вне зависимости от формата становится на середину рабочей области с измененным масштабом и осями координат. После этого модель готова к отправке в *CuraEngine* за упаковкой всех данных в класс *ArcusMessage*.

$$\begin{bmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Рис. 5. Матрицы преобразования:

a – матрица масштабирования; b – матрица перемещения

В случае, если режим печати выбран не «*classic*», то изменяется структура подготовки сообщения перед отправкой в сокет, так как будет использоваться другой движок нарезки. Из-за его особенностей уже необходимо инициализировать другие типы сообщений, ввести новые настройки печати и изменять команды G-кода, например, координатную систему принтера.

Данная разрабатываемая система позволит операторам 3D-принтеров не быть привязанным к определенному рабочему месту со специализированным программным обеспечением. Она помогает готовить модель к печати и отправлять ее на печать с любого устройства с доступом в интернет, хоть с телефона. Из дополнительных положительных моментов можно отметить то, что у конечных пользователей пропадает необходимость самостоятельной установки обновлений программ, потому что все последние изменения как сервера, так и движка нарезки всегда будут производиться на сервере. За счет отсутствия графического интерфейса (при взаимодействии компонентов сервера) повышается скорость нарезки, но не пропадает ее информативность. Также система является гибкой, в нее легко можно добавить новые режимы печати и привязать слайсеры моделей.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Особенности развития 3D печати и возникающие при этом проблемы / У. Ф. Сухих, Т. Э. Замалеев, Ф. Ф. Ахметсафина, М. Ф. Шаехов // Новые технологии и материалы легкой промышленности : матер. XVIII Всерос. науч.-практ. конф. с элементами научной школы для студентов и молодых ученых, Казань, 16–20 мая 2022 года. – Казань : Казанский национальный исследовательский технологический университет, 2022. – С. 108–110. – EDNYEXZEN.
2. Модуль подготовки деталей к печати в программном комплексе «Виртуальный 3D-принтер» / Е. А. Синицин, О. И. Бритова, В. В. Попов [и др.] // Молодежь в науке : сб. докл. 19-й науч.-техн. конф., Саратов, 09–11 ноября 2021 года. – Саратов : ФГУП «Российский федеральный ядерный центр – Всероссийский научно-исследовательский институт экспериментальной физики», 2022. – С. 150–156. – DOI 10.53403/9785951505200_150. – EDN LFUSZA.
3. Grechukhin, A. N. Model of Filling the Internal Structure of Workpiece with Curved Layers for 3D Printing / A. N. Grechukhin, V. V. Kuts // Materials Research Proceedings, Temryuk, 06–10 сентября 2021 года. – Temryuk, 2022. – P. 317–322. – DOI 10.21741/9781644901755-56. – EDN VPXMYU.