

УДК 004.9

DOI 10.35211/1990-5297-2020-9-244-22-25

*А. К. Гельштейн¹, А. М. Кашевник²***АВТОМАТИЗАЦИЯ ПРОЦЕССА ОБСЛУЖИВАНИЯ СТАНКА
КОЛЛАБОРАТИВНЫМ РОБОТОМ НА ОСНОВЕ
ШАБЛОННО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ***¹ Университет ИТМО² СПИИРАН

alexey.kashevnik@iias.spb.su

В статье представлен подход к автоматизации процесса обслуживания станка коллаборативным роботом, заключающийся в генерации программного кода для робота. Подход основывается на применении шаблонно-ориентированного программирования и позволяет учитывать как различные сценарии обработки деталей, так и входные параметры.

Ключевые слова: взаимодействие робототехнических систем, автоматизация, управление роботом, станок с ЧПУ.

*A. K. Gelshtein¹, A. M. Kashevnik²***MACHINE SERVICE AUTOMATION BY A COLLABORATIVE
ROBOT BASED ON PATTERN-ORIENTED PROGRAMMING**¹ ITMO University² SPIIRAS

The paper presents an approach to machine service process automation by a collaborative robot aimed at software code generation for the robot. The approach is based on the pattern-oriented programming and allows to take into account both various scenarios of part processing and input parameters.

Keywords: robotic systems interaction, automation, robot control, CNC machine.

В эпоху стремительного совершенствования технологий и усиления конкуренции производство стремится улучшить качество продукции и повысить производительность. Компании постоянно находятся в поиске способов внедрения инноваций для поддержки конкурентоспособности [1, 2]. Необходимо использование на всех этапах производственного процесса эффективных и технологичных решений [3], в том числе эффективных способов программирования роботизированных систем. Внедрение таких технологий [4] возможно в любых отраслях: автомобильная, пищевая, фармацевтическая и химическая; производство пластиков и полимеров, а также в любых производственных процессах, где следует исключить человеческий фактор, влияющий на конечный выход продукции.

В статье представлен подход к автоматизации процесса обслуживания станка коллаборативным роботом, основанный на применении шаблонно-ориентированного программирования. Подход был апробирован при внедрении коллаборативного робота UR10 фирмы Universal Robots в производство на предприятии в Санкт-Петербурге, изготавливающем датчики давления, уровня и температуры для газовой, нефтя-

ной и атомной отраслей. Робот UR10 используется для обслуживания станка с интегрированным ЧПУ фирмы HAAS и автоматизации процесса [5] на основе шаблонно-ориентированного моделирования сценариев.

Целью автоматизации производственного процесса является возможность генерации кода для робота, в зависимости от выбранного сценария. Сценарий может учитывать несколько видов станков, на которых происходит обработка, количество патронов и их ориентацию на столе станка, количество деталей для обработки и их высоту, а также последовательность всех манипуляций робота.

Для этого были разработаны шаблоны в виде программного кода, которые отвечают за определенные манипуляции робота. Далее, в зависимости от деталей, оператор составляет из этих шаблонов сценарий, по которому будет происходить доставка заготовок в станок. На выходе генерируется программный код, который управляет роботом для реализации заданного сценария.

Данный подход позволил сэкономить время разработки программного кода при смене заготовок, поступающих на станок.

В результате анализа предметной области были определены следующие функциональные требования к системе автоматизации процесса обслуживания станка коллаборативным роботом:

возможность просмотра всех шаблонов, внесенных в систему;

добавление в систему новых шаблонов по мере появления новых заготовок для обработки;

запись сгенерированного обеспечения в отдельный файл;

системные комментарии по ходу составления сценария для робота.

Также были определены нефункциональные требования:

должно быть ограничение на вводимое значение высоты заготовки;

система должна быть интуитивно понятной и удобной;

расширяемость – возможность использо-

вания большего количества и видов станков и манипуляторов;

производительность – быстрая генерация программного кода;

удобство сопровождения.

На рис. 1 представлена концептуальная модель системы автоматизации процесса обслуживания станка коллаборативным роботом. Информационное пространство, состоит: из набора сценариев, набора шаблонов, языка описания шаблонов – URScript и языка описания сценариев. На языке URScript разработаны шаблоны для возможных алгоритмов действия робота в различных условиях. Например, если на станке установлен один патрон, то в программном коде будет использоваться шаблон, описывающий действия робота в данной ситуации, или, если заготовку необходимо будет перевернуть для обработки другой стороны, также будет применен шаблон для этого случая.

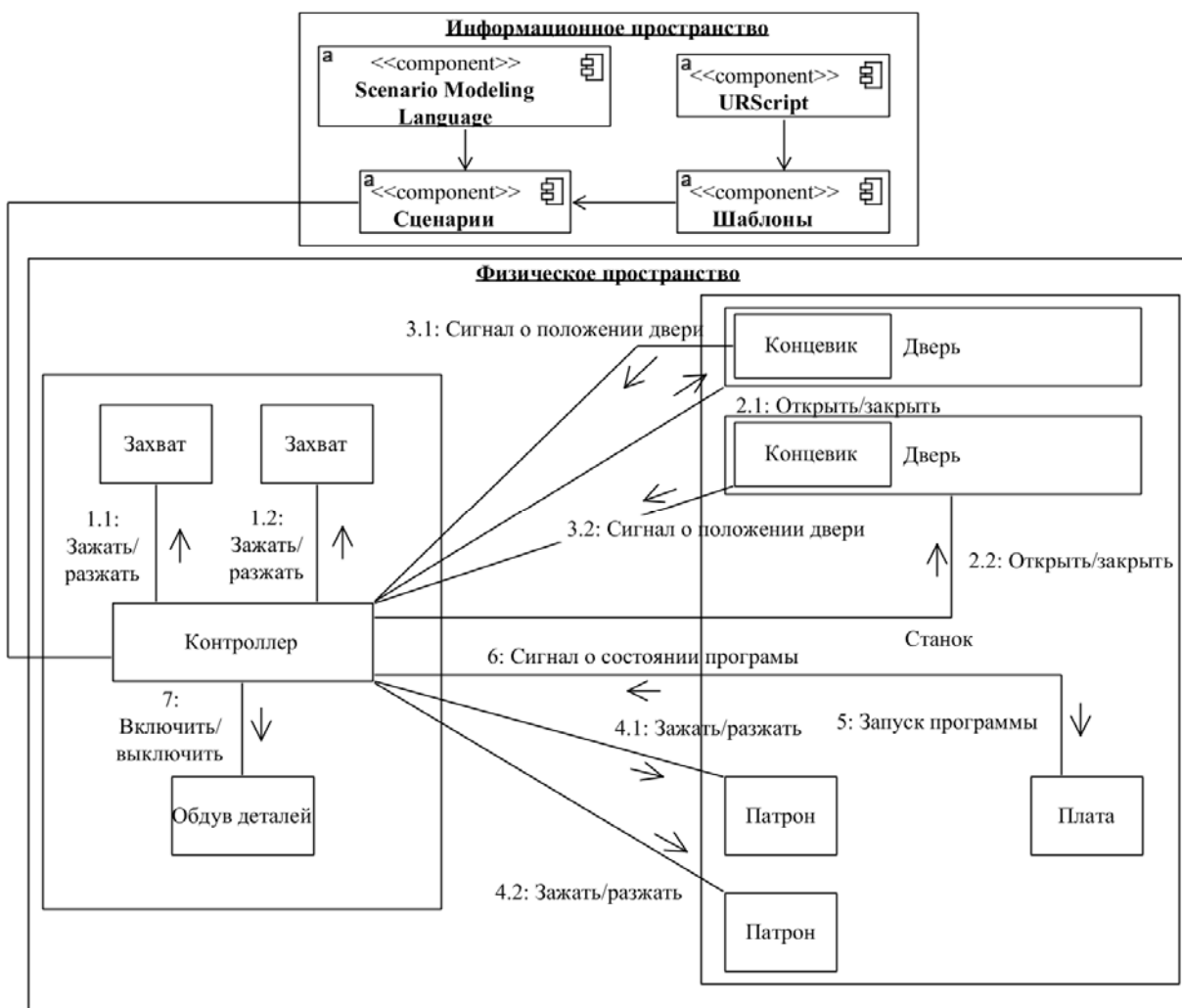


Рис. 1. Концептуальная модель системы автоматизации процесса обслуживания станка коллаборативным роботом

Физическое пространство представляет собой схему взаимодействия элементов робота и станка с сигналами, с помощью которых они управляются. На рис. 1 представлена общая схема роботизированного комплекса (количество патронов, дверей и захватов может меняться в зависимости от вида станка и заготовки, которая будет на нем обрабатываться).

Элементы управляются сигналами цифровых входов/выходов общего назначения контроллера. Например, цифровой выход управляет пневматическим реле для открытия/закрытия захвата на работе или включения/выключения функции обдува заготовок.

Основными элементами управления на станке являются двери, патроны и плата M08. Двери оснащены пневматическими цилиндрами для автоматического открытия сигналами от робота, а также концевиками для отслеживания их положения. Патроны, как и захваты, управляются пневматической системой для зажатия/разжатия. Плата M08 установлена в станок для возможности запуска цикла обработки с внешнего источника (робота), а также для контроля состояния программы с помощью ре-

ле. Для генерации сценариев был разработан язык описания сценариев (см. рис. 2) на основе шаблонов.

Каждому шаблону (Template) привязан программный блок (Program), разработанный на URScript. Пользователь «собирает» программу в соответствии со сценарием (Scenario) из необходимых шаблонов (Template), вводит требуемые входные данные и на выходе получает программный код (Script) для робота по описанному сценарию.

Алгоритм движения рабочего инструмента робота в целом складывается в задачу доставки детали из зоны складирования на паллете в зону обработки в станке. При этом алгоритмы движения имеют однотипный характер для разных деталей и могут быть постоянными, если в начале программирования ввести корректировку по высоте детали. Выбор шаблонов обуславливается тем, что общий цикл манипуляций робота разбивается на короткие и простые последовательности действий, которые в дальнейшем можно будет повторно использовать в других программах вне зависимости от обрабатываемой детали.

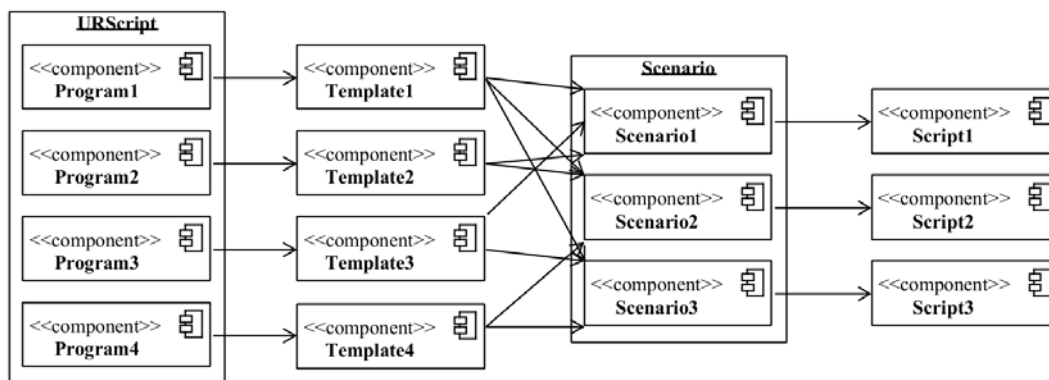


Рис. 2. Создание сценариев на основе шаблонов

Первой входной точкой в программе является выбор станка, на котором будет обрабатываться деталь. Для токарного станка набор шаблонов меньше, чем для фрезерного, так как место установки детали в нем – неизменная величина. Соответственно, могут меняться высота детали, их количество, и возможен переворот детали для обработки другой поверхности. Итого, получаются четыре шаблона:

1. Доставка детали от места складирования на паллете до зоны обработки.

2. Подача соответствующих сигналов для закрытия двери и запуска программы обработки и взятие новой детали.

3. Подача соответствующих сигналов для закрытия двери и запуска программы обработки, ожидания завершения цикла, открытие дверей, переворот детали, далее – п. 2.

4. Открытие двери, замена обработанной детали на заготовку, подача соответствующих сигналов для закрытия двери и запуска программы обработки, доставка детали до места складирования на паллете.

Помимо вышеописанных блоков в программе необходимо будет указать высоту детали и их количество на паллете. После этого будет сгенерирован скрипт для робота. В случае с фрезерным станком количество шаблонов

увеличивается, так как становится больше вариантов алгоритмов для робота. Например, может быть один патрон или два, или второй может быть установлен горизонтально. Помимо вышеописанных шаблонов также были разработаны два шаблона для каждого из видов станков. Для этих шаблонов происходит начальная установка полезной нагрузки и центра масс, объявляются переменные. Приложение для генерации сценариев реализовано на Python 3 и используется для взаимодействия с шаблонами, описанными выше. Для разработки интерфейса (рис. 3) была использована библиотека PySimpleGUI.

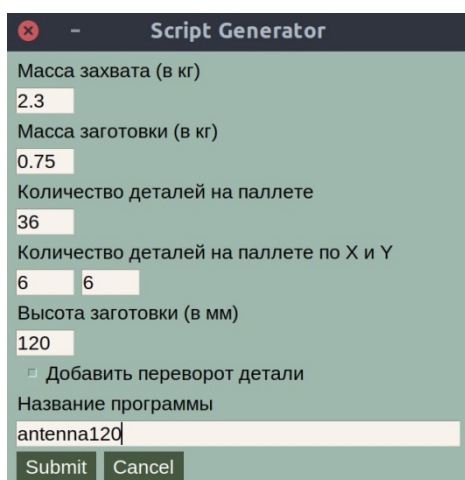


Рис. 3. Интерфейс приложения для генерации сценариев

Приложение использует шаблоны, на основе которых происходит генерация программного кода для управления роботом. Задание высоты заготовок, их массы (и прочее) происходит с использованием переменных, записанных в шаблонах.

После разработки самих шаблонов необходимо было их отформатировать. Параметры, которые требуют корректировки при генерации новых сценариев, были заменены на переменные.

```
d = dict(
    GRAB_WEIGHT=float(values['grabWeight']),
    DETAIL_WEIGHT=float(values['detailWeight']),
    NUMBER_OF_DETAILS=int(values['numberOfDetails']),
    NUMBER_OF_X_DETAILS=int(values['Xnumber']),
    NUMBER_OF_Y_DETAILS=int(values['Ynumber']),
    HEIGHT=(0.2 + float(values['height']) / 1000),
    NAME=str(values['name'])
)
```

Рис. 4. Список с переменными

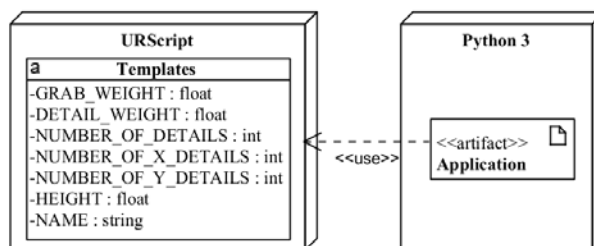


Рис. 5. Использование переменных в шаблонах

На рис. 4 представлен список переменных, значение которых задается в окне приложения. На рис. 5 приведена диаграмма взаимодействия приложения и переменных в шаблонах. После генерации сценария из шаблонов происходит присвоение переменным их значений с помощью парсера, который проходит весь документ. Для этого в приложении хранится список (рис. 4).

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Schneider, G. F. Knowledge-based Conversion of Finite State Machines in Manufacturing Automation / G. F. Schneider, G. A. Peßler, W. Terkaj // Procedia Manufacturing. – 2019. – Т. 28. – С. 189–194.
2. Bashiri, M. Design of a manufacturing hub with modular automation units / M. Bashiri, A. Golkarian, E. Nikzad // Procedia Manufacturing. – 2018. – Т. 17. – С. 911–918.
3. Seder, M. et al. Open Platform Based Mobile Robot Control for Automation in Manufacturing Logistics // IFAC-PapersOnLine. – 2019. – Т. 52. – №. 22. – С. 95–100.
4. Zheng, C. et al. SME-oriented flexible design approach for robotic manufacturing systems // Journal of Manufacturing Systems. – 2019. – Т. 53. – С. 62–74.
5. Schäffer, E. et al. Microservice-based architecture for engineering tools enabling a collaborative multi-user configuration of robot-based automation solutions // Procedia CIRP. – 2019. – Т. 86. – С. 86–91.