

РАЗРАБОТКА СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ
АЛГОРИТМОВ УПРАВЛЕНИЯ ОБЪЕКТАМИ АВТОМАТИЗАЦИИ



И. А. Макаренко, студент группы 4-го курса
E-mail: ilyamakarenko0703@gmail.com
ФГБОУ ВО «Калининградский государственный
технический университет»

Н. А. Долгий, канд. техн. наук, доцент
E-mail: dolgi@klgtu.ru
ФГБОУ ВО «Калининградский государственный
технический университет»

В статье предложена система автоматизированного проектирования алгоритмов управления объектами автоматизации, основной задачей которой является автоматизированная генерация программного кода на основе визуально описанных алгоритмов управления. Ключевым элементом разработки является создание программного инструмента, обеспечивающего преобразование графов состояний, разработанных инженером-проектировщиком, в программные модули, совместимые со средами промышленного программирования, соответствующими стандарту ИЕС 61131-3. На примере объекта газотурбинной компрессорной установки ГТК-10-4 рассмотрен практический процесс применения разработанного инструмента: от анализа и верификации полученных алгоритмов управления до генерации программного кода.

Ключевые слова: *алгоритм, граф состояний, газотурбинная компрессорная установка, конечный автомат, программный код*

ВВЕДЕНИЕ

Современные системы автоматизации играют ключевую роль в повышении эффективности и надежности технологических процессов в различных отраслях промышленности. Одним из ключевых элементов таких систем являются алгоритмы управления, которые обеспечивают выполнение задач управления в соответствии с заданными требованиями [1]. Алгоритмы управления сложными технологическими объектами обладают рядом особенностей, связанных со спецификой объекта управления, включая требования к безопасности, надежности и производительности [2]. Для автоматизированного проектирования алгоритмов управления используют сложные программно-технические комплексы, предназначенные для одновременной работы большого количества специалистов-проектировщиков [3]. Производственные технические системы, как правило, имеют сложную структуру, характеризующуюся высоким уровнем автоматизации технологических операций и операций управления, различным технологическим оборудованием, для управления которым используется сложное программное обеспечение (ПО).

Большое количество нелинейных, взаимосвязанных и измеряемых параметров контроля и регулирования усложняет разработку алгоритма управления объектом автоматизации. С целью упрощения процедуры разработки алгоритмов управления в статье рассматривается возможность использования системы автоматизированного проектирования

алгоритмов управления, которая на основе визуальных моделей реализует алгоритмы управления объектами автоматизации на языке программирования ST стандарта IEC 61131-3.

ОБЪЕКТ ИССЛЕДОВАНИЯ

Объектом исследования в данной работе являются алгоритмы управления для сложных технических систем. В качестве примера для демонстрации разрабатываемого подхода рассматривается газотурбинная компрессорная установка ГТК-10-4, широко используемая в энергетике и промышленности. Управление таким объектом связано с необходимостью обработки большого количества входных данных, обеспечения устойчивости работы в различных режимах и минимизации рисков аварийных ситуаций [4]. Сложность разработки алгоритмов управления для ГТК-10-4 заключается в необходимости учета многочисленных взаимосвязей между подсистемами, а также в обеспечении высокой степени детерминированности и предсказуемости поведения системы в целом.

ЦЕЛЬ И ЗАДАЧИ ИССЛЕДОВАНИЯ

Главной задачей данной работы является разработка системы автоматизированного проектирования алгоритмов управления системами автоматизации, ориентированной на поддержку инженеров-программистов в процессе разработки программных модулей ПО для системы автоматизированного управления. Основным отличием разрабатываемой системы от традиционных подходов является возможность генерации программного кода управляющих алгоритмов на основе визуальных алгоритмов управления. Кроме того, внедрение подобной системы имеет значительную экономическую составляющую, так как снижает затраты на разработку и поддержку программного обеспечения, а также минимизирует риски ошибок, которые могут привести к аварийным ситуациям и финансовым потерям.

Таким образом, актуальность данной работы обусловлена необходимостью повышения эффективности разработки алгоритмов управления для сложных технических систем, таких как ГТК-10-4, а также потребностью в создании инструментов, которые позволят автоматизировать и упростить процессы проектирования, верификации и сопровождения ПО.

МЕТОДЫ ИССЛЕДОВАНИЯ

Для решения задачи разработки алгоритмов управления в работе предлагается использовать подход, основанный на концепции конечных автоматов, графов и алгоритмов подсистем. Конечный автомат позволяет формализовать поведение системы, выделив основные состояния и переходы между ними, что упрощает анализ и проектирование [5], [6]. Графы состояний описывают работу подсистем, используются для визуализации и анализа взаимодействия между компонентами системы, что способствует более глубокому пониманию ее функционирования. Алгоритмы описывают задачи, которые решаются в рамках каждой подсистемы, что позволяет декомпозировать сложную задачу управления на более простые и понятные элементы.

РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЯ

Процесс создания систем автоматизации, включая алгоритмы управления, требует предварительного проектирования, на котором принимаются ключевые решения, влияющие на надежность, эффективность и масштабируемость конечного проекта. Особенно это актуально при разработке инструментальных средств, таких как системы автоматизированного проектирования, направленных на поддержку инженеров при формализации и реализации алгоритмов управления.

Разработка системы автоматизированного проектирования алгоритмов управления требует четкого разграничения ролей и эффективного взаимодействия между двумя специалистами, участвующими в проекте — инженером-проектировщиком и инженером-

программистом. Такое разделение обусловлено различием в профессиональных компетенциях и подходах к проектированию.

Инженер-проектировщик (ИПр) обладает углублёнными знаниями об объекте управления, его технологических характеристиках и особенностях функционирования. Однако, он не владеет методикой технологического программирования и не работает напрямую с инструментами разработки ПО. Инженер-программист (ИП), в свою очередь, компетентен в языках программирования, структуре программного кода и конфигурации систем управления, но не всегда имеет полное представление о логике и поведении управляемого объекта.

Для эффективной реализации проекта необходимо выстроить процесс взаимодействия таким образом, чтобы каждый специалист выполнял задачи в рамках своей области ответственности, обеспечивая при этом непрерывную и корректную передачу информации. В рамках предлагаемой структуры эта задача решается за счёт использования визуальных моделей алгоритмов управления, разрабатываемых ИПр в форме конечных автоматов и графов состояний. Такие модели являются интуитивно понятными и пригодными для последующей технической реализации. На рисунке 1 представлена схема взаимодействия ИПр и ИП при проектировании алгоритмов управления.

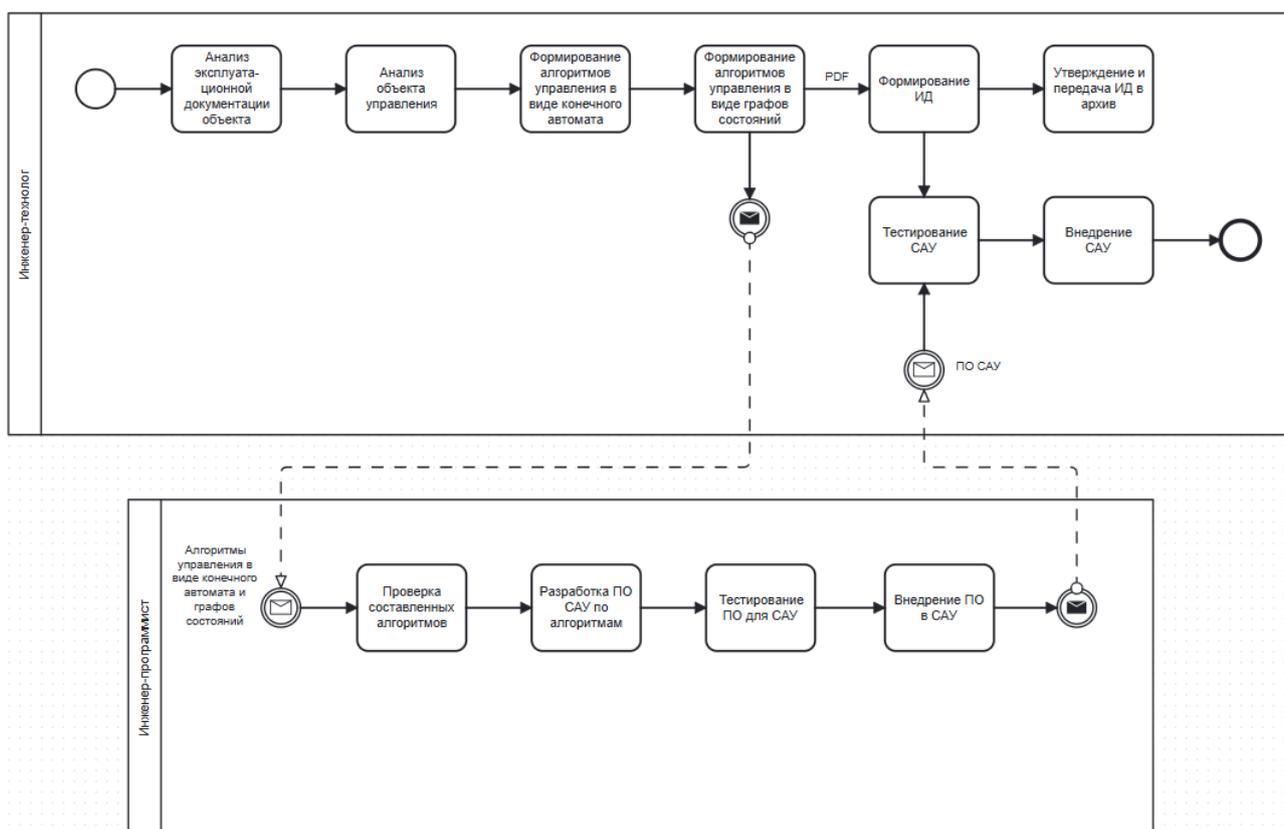


Рисунок 1 — Взаимодействие ИПр и ИП при проектировании алгоритмов управления

ИПр обладая знаниями об объекте управления, выполняет следующие действия: анализ эксплуатационной и технологической документации, анализ поведения объекта управления, формирование алгоритмов управления сначала в виде конечного автомата, затем в виде графов состояний (рисунок 2).

Таблица переходов		1	2	3	4	5	6	7	8	9	10
		АО	НО	Ремонт	КГП	ГП	АП	К	М	К-М	М-К
1	Аварийная остановка (АО)										
2	Нормальная остановка(НО)										
3	Ремонт										
4	Контроль готовности к пуску (КГП)										
5	Готовность к пуску (ГП)										
6	Авт. Пуск (АП)										
7	Кольцо (К)										
8	Магистраль (М)										
9	Кольцо-Магистраль (К-М)										
10	Магистраль-Кольцо (М-К)										

Рисунок 2 — Графов состояний

Следующим этапом заполняется таблица описания автомата состояний (рисунок 3). В данной таблице указываются условия переходов, по которым запускается выполнение определённого режима установки ГТК-10-4. В колонке таблицы «Формируемые сигналы» незаполненная строка означает статистическое состояние объекта.

№ пп	Переход		Условие перехода	Формируемые сигналы		
	Из	В				
1	1	Аварийная остановка (АО)	3	Ремонт	Завершение режима и деблокировка	
2	2	Нормальная остановка(НО)	1	Аварийная остановка (АО)	АО	Режим "АО"
3	2	Нормальная остановка(НО)	4	Контроль готовности к пуску (КГП)	Завершение режима	
4	3	Ремонт	4	Контроль готовности к пуску (КГП)	Команда "КГП"	
5	4	Контроль готовности к пуску (КГП)	1	Аварийная остановка (АО)	АО	Режим "АО"
6	4	Контроль готовности к пуску (КГП)	3	Ремонт	Команда "Ремонт"	
7	4	Контроль готовности к пуску (КГП)	5	Готовность к пуску (ГП)	Готовность к пуску	
8	5	Готовность к пуску (ГП)	1	Аварийная остановка (АО)	АО	Режим "АО"
9	5	Готовность к пуску (ГП)	3	Ремонт	Команда "Ремонт"	
10	5	Готовность к пуску (ГП)	4	Контроль готовности к пуску (КГП)	Отсутствие готовности к пуску	
11	5	Готовность к пуску (ГП)	6	Авт. Пуск (АП)	Команда "АП"	Режим "АП"
12	6	Авт. Пуск (АП)	1	Аварийная остановка (АО)	АО	Режим "АО"
13	6	Авт. Пуск (АП)	2	Нормальная остановка(НО)	Команда "НО"	Режим "НО"
14	6	Авт. Пуск (АП)	7	Кольцо (К)	Завершение режима	Режим "К"
15	7	Кольцо (К)	1	Аварийная остановка (АО)	АО	Режим "АО"
16	7	Кольцо (К)	2	Нормальная остановка(НО)	Команда "НО"	Режим "НО"
17	7	Кольцо (К)	9	Кольцо-Магистраль (К-М)	Команда "М"	Режим "К-М"
18	8	Магистраль (М)	1	Аварийная остановка (АО)	АО	Режим "АО"
19	8	Магистраль (М)	2	Нормальная остановка(НО)	Команда "НО"	Режим "НО"
20	8	Магистраль (М)	10	Магистраль-Кольцо (М-К)	Команда "К"	Режим "М-К"
21	9	Кольцо-Магистраль (К-М)	1	Аварийная остановка (АО)	АО	Режим "АО"
22	9	Кольцо-Магистраль (К-М)	2	Нормальная остановка(НО)	Команда "НО"	Режим "НО"
23	9	Кольцо-Магистраль (К-М)	8	Магистраль (М)	Завершение режима	Режим "М"
24	9	Кольцо-Магистраль (К-М)	10	Магистраль-Кольцо (М-К)	Команда "К"	Режим "М-К"
25	10	Магистраль-Кольцо (М-К)	1	Аварийная остановка (АО)	АО	Режим "АО"
26	10	Магистраль-Кольцо (М-К)	2	Нормальная остановка(НО)	Команда "НО"	Режим "НО"
27	10	Магистраль-Кольцо (М-К)	7	Кольцо (К)	Завершение режима	Режим "К"

Рисунок 3 — Таблица описания автомата состояний

В качестве примера рассмотрим одно из динамических состояний автомата – граф состояния режима автоматического пуска (АП) установки (рисунок 4). Данный алгоритм управления определяет последовательность пусковых операций подсистем ГТК-10-4. В левой части графа указаны условия переходов на следующий этап (задержки и проверки), а в правой части отображены действия (сигналы на автоматизированное рабочее место и запуск подрежимов работы системы).

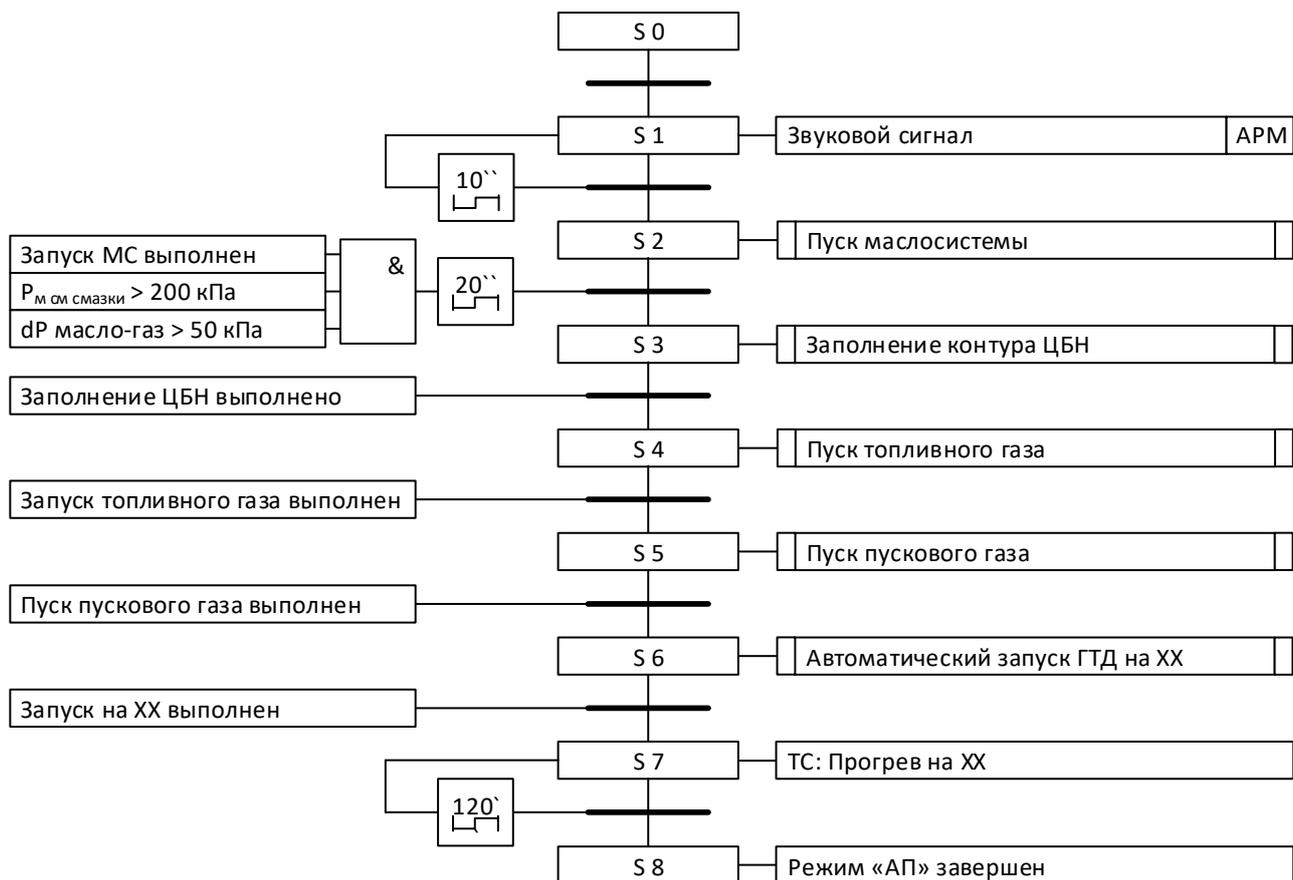


Рисунок 4 — Граф состояния режима автоматического пуска (АП) агрегата

Полученный граф описывает переходы между состояниями системы. Он служит формализованной основой для дальнейшей технической реализации алгоритмов управления. В рамках реализации автоматизированной системы проектирования алгоритмов управления важно обеспечить не только удобную визуализацию логики, но и её дальнейшую интерпретацию в виде исполнимого кода.

Основным входным элементом системы проектирования является файл, содержащий описание алгоритма в виде графа состояний. В качестве средства создания такого представления используется программное обеспечение Microsoft Visio, в котором алгоритм формализуется через набор взаимосвязанных фигур, отражающих состояния, переходы, действия и условия. Документ Visio содержит структурированную визуальную модель, соответствующую логике функционирования системы управления. Каждая фигура графа снабжается необходимыми атрибутами: идентификатором, типом, описанием действия, условием перехода и т.д. За счёт возможности чтения и анализа структуры Visio файлов с использованием языков программирования (в частности, Python с библиотекой ruwin32), такая форма представления может быть формально интерпретирована и преобразована в программный код. На рисунке 5 приведен инструмент генерации с использованием Python для реализации алгоритмов управления ГТК-10-4 на языке ST.

```

import win32com.client
from win32com.client import constants as c
import re
from openpyxl import load_workbook

def initializationDef(shape):
    text = shape.Text.strip().replace("`", "")
    if not text and shape.Shapes.Count == 0:

```

```

return
if shape.SectionExists(c.visSectionProp, 0):
    for row in range(0, 100):
        if not shape.RowExists(c.visSectionProp, row, 0):
            continue
        try:
            label = shape.CellsSRC(c.visSectionProp, row, c.visCustPropsLabel).ResultStr("")
            if 'Де́йстви́е' in label:
                wb = load_workbook("ГТК-10-4v1.xlsx", data_only=True)
                sheet = wb['Visio']
                row_w = 2
                while sheet.cell(row=row_w, column=1).value is not None:
                    if sheet.cell(row=row_w, column=1).value in text:
                        name = sheet.cell(row=row_w, column=2).value
                        if 'sm3' in name:
                            initialization.append(f" retSm[0] := {name} (SM_RESET, 1);")
                        elif 'alg' in name:
                            initialization.append(f" {name} := false;")
                        row_w += 1
                    if 'Задержка' in label or 'Задержка1' in label:
                        time.append("")
            except:
                continue

```

Рисунок 5 — Фрагмент программного кода Python с библиотекой ruwin32

ЗАКЛЮЧЕНИЕ

Разработанный инструмент генерации кода с использованием Python для реализации системы автоматизированного проектирования алгоритмов управления систем автоматизации на примере газотурбинной компрессорной установки ГТК-10-4 осуществляет обработку входного файла, а именно считывает структуру графа состояний из Visio-документа, извлекает ключевую информацию об элементах алгоритма, выполняет преобразование графовой модели в текстовый код на языке ST, соответствующий стандарту IEC 61131-3 и сохраняет результат в виде файла совместимого с конфигурацией САУ. На выходе системы информационного обеспечения формируется частично или полностью готовый программный модуль, который может быть использован в дальнейшем для загрузки в контроллер или интеграции в проект системы автоматизации установки ГТК-10-4.

СПИСОК ЛИТЕРАТУРЫ

1. Модели и алгоритмы концептуального проектирования автоматизированных систем управления / Д. В. Богданов [и др.]; [под ред. С. Г. Чекинова]. – Москва: Компания Спутник+, 2004. – 323 с. – ISBN 5-93406-793-1. – EDN OVRPXR.
2. Лях, Т. В. Автоматическая верификация алгоритмов управления сложными технологическими объектами на программных имитаторах / Т. В. Лях, В. Е. Зюбин// Математическое и компьютерное моделирование: Сборник материалов V Международной научной конференции, посвященной памяти Р.Л. Долганова, Омск, 01 декабря 2017 года / Ответственный за выпуск И.П. Бесценный. – Омск: Омский государственный университет им. Ф.М. Достоевского, 2017. – С. 128-130. – EDN YOKPQL.

3. Айрапетов, Д. А. К вопросу об управлении процессом проектирования в рамках САПР СУ сложными технологическими объектами / Д. А. Айрапетов, Г. Г. Арунянц // Инновации в науке, образовании и бизнесе - 2012: Труды X Международной научной конференции. В 2-х частях, Калининград, 17–19 октября 2012 года. Том Часть 2. – Калининград: Калининградский государственный технический университет, 2012. – С. 148-150. – EDN ZLGTCSB.
4. Персидский, А. В. Автоматизированная система управления установкой подготовки газа / А. В. Персидский, В. Н. Шерстобитова // Компьютерная интеграция производства и ИПИИ-технологии: Материалы XI Всероссийской конференции, Оренбург, 16 ноября 2023 года. – Оренбург: Оренбургский государственный университет, 2023. – С. 292-294. – EDN XDIMFI.
5. Царев, Ф. Н. Методы построения конечных автоматов на основе эволюционных алгоритмов: специальность 05.13.11 "Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей": диссертация на соискание ученой степени кандидата технических наук / Царев Федор Николаевич. – Санкт-Петербург, 2012. – 196 с. – EDN QGAYAN.
6. Кудрявцев, М. А. О подходе к генерации тестовых данных для эвристических алгоритмов обработки конечных автоматов / М. А. Кудрявцев // Прикладная математика и информатика: современные исследования в области естественных и технических наук: Материалы IV научно-практической международной конференции (школы-семинара) молодых ученых: в 2 частях, Тольятти, 23–25 апреля 2018 года. Том Часть 2. – Тольятти: Качалин Александр Васильевич, 2018. – С. 168-172. – EDN UZTLKD.

DEVELOPMENT OF A SYSTEM FOR AUTOMATED DESIGN OF ALGORITHMS FOR CONTROLLING AUTOMATION OBJECTS

I.A. Makarenko,
4th year student
e-mail: ilyamakarenko0703@gmail.com
Kaliningrad State Technical University

N.A. Dolgii,
PhD in Engineering
e-mail: dolgi@klgtu.ru
Kaliningrad State Technical University

This article proposes a system for automated design of control algorithms for automation objects, the main task of which is the automated generation of program code based on visually described control algorithms. A key element of the development is the creation of a software tool that enables the transformation of state graphs developed by a process engineer into software modules compatible with industrial programming environments that comply with the IEC 61131-3 standard. Using the example of the GTK-10-4 gas turbine compressor unit, the practical process of using the developed tool is considered: from analysis and verification of the obtained control algorithms to the generation of program code.

Keywords: *algorithm, state graph, gas turbine compressor unit, finite state machine, program code*