

5. Пегат А. Нечеткое моделирование и управление/ А. Пегат. – М.: Бином. Лаборатория знаний, 2013. – 798 с.

6. Mamdani E.H. Application of Fuzzy Logic to Approximate Reasoning Using Linguistic Synthesis // IEEE, Transactions on Computers. – 1977. – Vol. C-26. – №. 12. – P. 1182–1191.

УДК 004.896 : 621.316

В.Н. Дубинин¹, А.В. Дубинин², А.С. Войнов³, И.В. Сенокосов⁴, В. Вяткин⁵

**ПРЕОБРАЗОВАНИЕ ОПИСАНИЙ ЦИФРОВЫХ ПОДСТАНЦИЙ
МЭК 61850 В МОДЕЛИ ПЕРЕХОДОВ СОСТОЯНИЙ СИСТЕМ
ЗАЩИТЫ С ИСПОЛЬЗОВАНИЕМ ОНТОЛОГИЙ**

Пензенский государственный университет

(Россия, Пенза, ¹dubinin.victor@gmail.com, ²dubinin.aleksey@gmail.com,

³voj49@yandex.ru, ⁴senokosov.i@yandex.ru)

Технический университет Лулео

(Швеция, Лулео, ⁵valeriy.vyatkin@ltu.se)

Аннотация. Представлен подход к преобразованию исходных описаний цифровых подстанций МЭК 61850 в модели систем защиты от перегрузок. В рамках подхода разработана онтология трансформации, включающая онтологию паттернов поиска, онтологию шаблонов применения и онтологию схем трансформационных правил. Правила трансформации распределены по семантическому Web, причем паттерны поиска, шаблоны применения и схемы правил могут находиться на различных серверах сети, чем достигается открытость, гибкость и повторная используемость решений. Выделены различные подходы к построению паттернов поиска, различающиеся способом представления паттерна (с помощью классов OWL, с помощью экземпляров OWL или с использованием языка SWRL), а также расположением паттерна (в виде отдельной онтологии или внутри общей онтологии подстанции). В качестве основной целевой модели используются конечные автоматы. Особенностью онтологии конечно-автоматной модели является использование сторожевых условий переходов. В качестве примера рассмотрена конечно-автоматная модель защиты фидера от перегрузок. Предложены архитектурные варианты организации схем защиты от перегрузок на основе функциональных блоков МЭК 61499, среди которых топология с произвольными связями, звездообразная топология, смешанная топология: Предложен набор программных средств для поддержки заявленного подхода. Основными программами, требующими разработки, являются трансляторы, осуществляющие преобразование OWL- и Prolog-базированных форм представления онтологий, линковщик онтологий и интерпретатор правил трансформаций на языке Prolog. Предложенный подход может быть использован в проектировании систем автоматизации цифровых подстанций МЭК 61850 на различных этапах.

Ключевые слова: сети электроснабжения, цифровая подстанция, защита от перегрузок, стандарт МЭК 61850, онтология, паттерн поиска, трансформация, стандарт МЭК 61499, функциональный блок, САПР.

V.N. Dubinin¹, A.V. Dubinin², A.S. Voinov³, I.V. Senokosov⁴, V. Vyatkin⁵
**TRANSFORMATION OF IEC 61850 SUBSTATIONS DESCRIPTIONS
TO STATE TRANSITION MODELS OF SUBSTATIONS PROTECTION
USING ONTOLOGIES**

Penza State University

(Russia, Penza, ¹dubinin.victor@gmail.com, ²dubinin.aleksey@gmail.com,
³voj49@yandex.ru, ⁴senokosov.i@yandex.ru)

Lulea University of Technology

(Sweden, Lulea, ⁵valeriy.vyatkin@ltu.se)

Abstract. An approach is presented to transform source descriptions of IEC 61850 substations to models of overload protection systems. As part of the approach, an ontology of transformation has been developed, including an ontology of search patterns, an ontology of application templates, and an ontology of transformation rules schemes. The transformation rules are distributed over the semantic Web, so search patterns, application templates and rule schemes can be located on different servers of the network. That results in openness, flexibility and reusability of solutions. Different approaches to building search patterns are recognized, differing in the way the pattern is presented (using OWL classes, using OWL individuals, or using the SWRL language), as well as the pattern location (as a separate ontology or within a general ontology of a substation). Finite automata are used as the main target model. A feature of the ontology of the finite automaton is the use of guard conditions for transitions. As an example, a finite automaton-based model of feeder protection against overloads is considered. Architectural options for organizing overload protection schemes based on IEC 61499 function blocks, including a topology with arbitrary connections, a star topology, a mixed topology, are proposed: A toolchain has been introduced to support the stated approach. The main programs requiring development are translators which transform OWL- and Prolog-based ontology representation, a linker of ontologies and an interpreter of transformation rules in Prolog. The proposed approach can be used in the design of automation systems for IEC 61850 substations at various stages.

Keywords: power supply network, substation, overload protection, IEC 61850 standard, ontology, search pattern, transformation, IEC 61499 standard, function block, CAD.

Введение

В настоящее время все большее распространение получают умные сети электроснабжения (*Smart grid*), обладающие более высокими технико-экономическими показателями по сравнению с традиционными [1,2]. Большую роль для построения сетей *Smart grid* играет международный стандарт МЭК 61850. Это универсальный стандарт, который позволяет упорядочить разрозненные решения различных производителей устройств релейной защиты и систем передачи данных, применяемых на цифровых подстанциях (ЦПС). Схемы защиты позволяют избежать остановки производственных или других процессов, вызванной критическим изменением напряжения или ошибками короткого замыкания. В ряде

зарубежных работ было показано, что наиболее адекватным средством для реализации схем защиты являются функциональные блоки (ФБ) стандарта МЭК 61499. На данный момент существует весьма незначительно число публикаций, посвященных разработке систем защиты цифровых подстанций МЭК 61850. Одно из таких исследований – это работа [2], в которой представлен общий подход для автоматической генерации программного обеспечения систем автоматизации *Smart grid*. В качестве целевой модели там рассматриваются системы управления защитой от перегрузок на основе ФБ МЭК 61499. Подход основан на трансформации онтологий с помощью языка *eSWRL*. Моделирование подобных систем также рассмотрено в [3].

Представленная нами работа является развитием работы [2] в направлении повышения гибкости и открытости подхода, повышения степени повторного использования решений. Это достигается тем, что правила трансформации распределены по семантическому Web, причем паттерны поиска, шаблоны применения и схемы правил могут находиться на различных серверах сети, чем достигается открытость, легкая реконфигурируемость и простота использования готовых решений.

Структура онтологии трансформации

На рис. 1 представлена структура онтологии трансформации. Как видно из данного рисунка, она состоит из трех частей: онтологии паттернов поиска, онтологии шаблонов применения и онтологии схем правил. Паттерны поиска представляют левую часть правил трансформации, в то время как шаблоны применения - правую часть. Связывание левых и правых частей в некоторое завершенное правило производится с помощью онтологии схем правил.

Онтологии паттернов поиска

Онтология паттернов поиска используется для поиска фрагментов электрических цепей в описаниях ЦПС МЭК 61850 для применения шаблонов схем защиты. Классификацию подходов к построению паттернов поиска для конечного пользователя можно производить на основе следующих атрибутов: атрибут А – способ представления паттерна (А1 – как Т-Box; А2 – как А-Box (+Т-Box); А3 – с использованием языка SWRL); атрибут В – отдельный/встроенный онтологический паттерн (В1 – паттерн в виде отдельной онтологии; В2 – паттерн внутри общей онтологии). Введем следующие обозначения подходов: А1В1 («отдельный паттерн, только Т-Box»; А2В1 («отдельный паттерн, Т-Box+А-Box»; А3В1 («отдельный паттерн, SWRL»); А2В2 («паттерн А-Box внутри общей онтологии ЦПС МЭК 61850»). Ниже представленные подходы рассматриваются более подробно. Для

упрощения примеров будет использоваться следующее подмножество классов: *Bay* – группа связанных устройств, которые образуют законченный элемент ЦПС; *ConductingEquipment* – оборудование сети; *VoltageLevel* – элементы подстанции, которые работают с одним напряжением. Объектные свойства: *has_VoltageLevel*, *has_Bay*, *has_ConductingEquipment*. Свойства по данным: *has_name*, *has_type*.

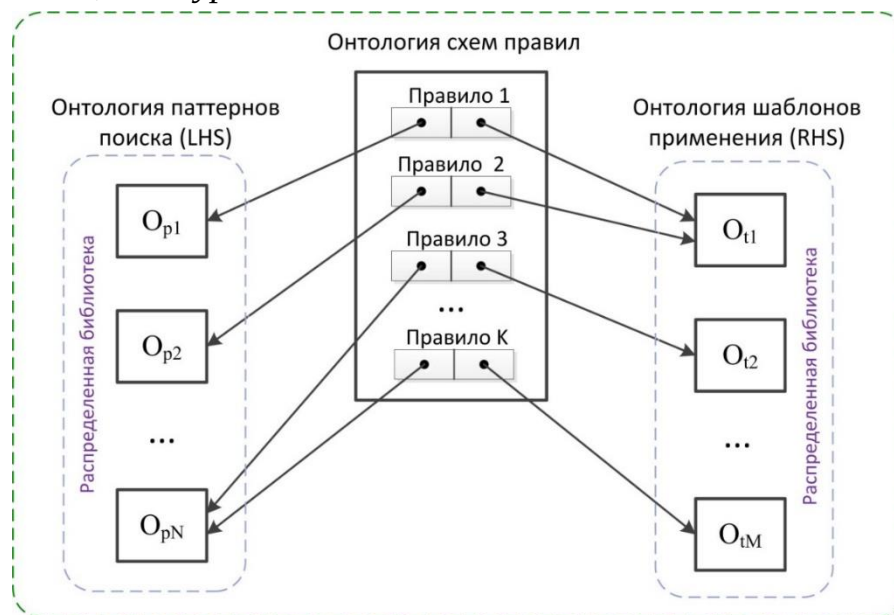


Рис. 1. Онтология трансформации

Подход A1B1

Принципы построения онтологии паттерна: 1) используются только необходимые классы и свойства основной онтологии ЦПС; 2) для прямого использования создаются подклассы и подсвойства соответствующих классов и свойств основной онтологии; 3) экземпляры классов не используются в принципе; 4) Знак "_" используется для индикации подсвойств по данным; 5) Знак "@" используется для обозначения переменных на местах значений свойств по данным. На рис. 2 в нижней части приведен пример паттерна и его взаимосвязи с классами основной онтологии ЦПС. На рис. 3 в качестве примера представлено описание класса *bayCE2* в в плагине Ontograf системы Protégé. На данном рисунке запись *@bayCE2Name* означает переменную.

Подход A2B1

В отличие от подхода A1B1 в подходе A2B1 паттерн в основном определяется как множество взаимосвязанных экземпляров (индивидуалов). На рис. 4 представлен пример паттерна (внизу) и его взаимосвязь с классами, а на рис. 5 – описание экземпляра *bayCE2* класса *Bay* в Protégé.

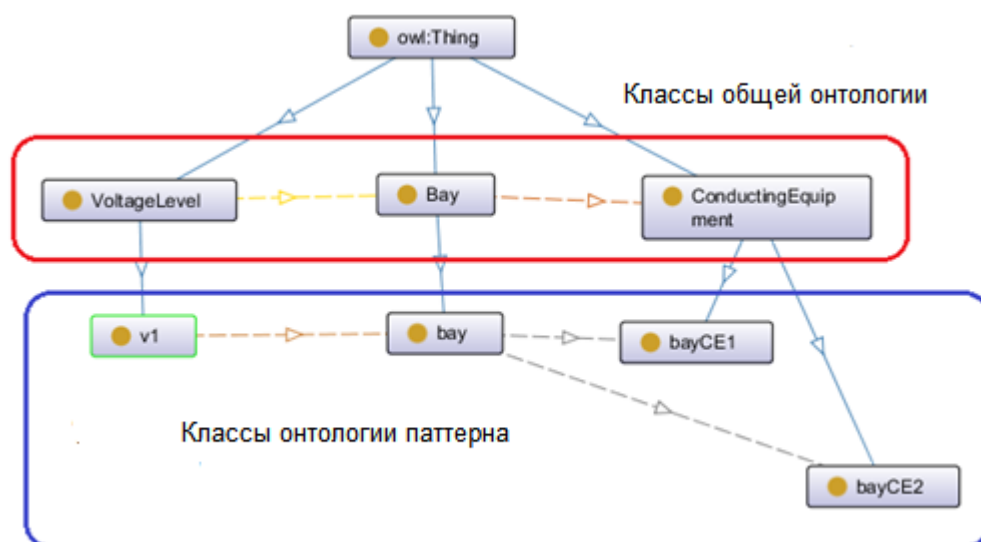


Рис. 2. Представление паттерна в системе Protégé (подход A1B1)

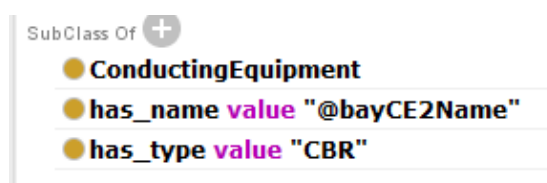


Рис. 3. Описание класса bayCE2 в онтологии паттерна

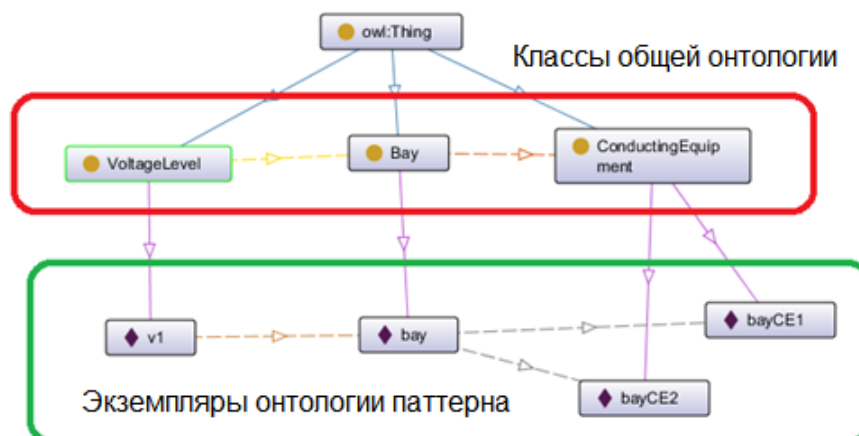


Рис. 4. Представление паттерна в системе Protégé (подход A2B1)

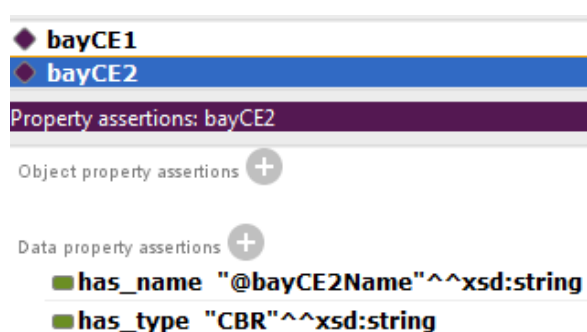


Рис. 5. Описание экземпляра bayCE2 в онтологии паттерна

Подход АЗВ1

В данном случае паттерн для поиска представляется левой частью SWRL-правила. В приведенном ниже примере правая часть SWRL-правила используется только для тестирования.

Bay(?bay), VoltageLevel(?v1), ConductingEquipment(?bayCE1), has_Bay(?v1, ?bay), DifferentFrom (?bayCE1, ?bayCE2), has_name(?bayCE1, ?bayCE1Name), ConductingEquipment(?bayCE2), has_name(?bayCE2, ?bayCE2Name), has_type(?bayCE2, "CBR"), has_type(?bayCE1, "CTR") -> ResultInst(?bayCE1), ResultInst(?bayCE2)

Онтология конечно-автоматной модели

Для представления моделей систем защиты ЦПС МЭК 61850 могут использоваться модели переходов состояний: конечные автоматы, формальные грамматики, сети Петри и т.д. В данной работе выбрана конечно-автоматная (КА) модель ввиду ее удобства, наглядности и широкого использования в целевой предметной области. Из особенностей используемой КА-модели можно отметить использование сторожевых условий. В табл. 1 представлена разработанная онтология T-Vox КА-модели, а на рис. 6 в качестве примера приведена известная КА-модель защиты фидера от перегрузок.

Таблица 1. Онтология T-Vox конечно-автоматной модели в Protege

Classes	Object Properties	Datatype Properties
<ul style="list-style-type: none"> owl:Thing <ul style="list-style-type: none"> Guard Event <ul style="list-style-type: none"> Event_output Event_input State <ul style="list-style-type: none"> Final_state Initial_state Transition 	<ul style="list-style-type: none"> owl:topObjectProperty <ul style="list-style-type: none"> has_guard has_output_event has_input_event from_state to_state 	<ul style="list-style-type: none"> owl:topDataProperty <ul style="list-style-type: none"> has_guard_expression has_name has_source

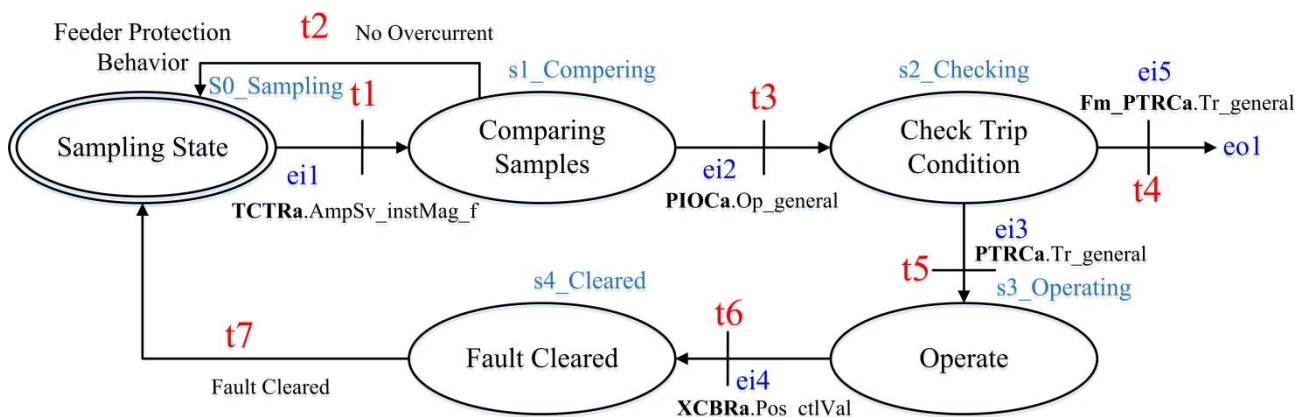


Рис. 6. Конечно-автоматная модель защиты фидера от перегрузок

Организация и структура системы защиты в ЦПС МЭК 61850

Для защиты ЦПС от перегрузок используются логические узлы TCTR, PIOC, PTRC, XCBR [2]. При их реализации на ФБ используется отображение «один к одному».

Предлагается несколько архитектурных подходов к организации схем защиты на ФБ, а именно: топология с произвольными связями, звездообразная топология, смешанная топология. Звездообразная топология представлена на рис. 7.

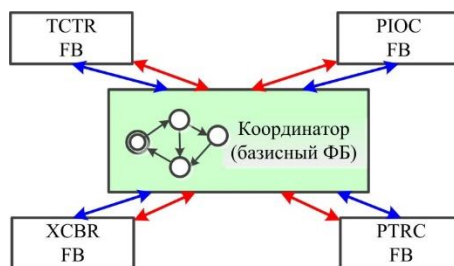


Рис. 7. Звездообразная архитектура

В данной архитектуре все интеракции и коммуникации между ФБ, представляющими логические узлы, осуществляются через центральный координатор (диспетчер) защиты, который может быть выполнен в виде или базисного ФБ, или составного ФБ.

Преимущества архитектуры: минимальный семантический разрыв между схемой защиты в форме КА и реализацией, легкость отображения схемы защиты на систему ФБ; возможность использования общего хранилища данных и средств вычислений.

Инструментарий

На рис. 8 показана структура программных средств для поддержки предложенного выше подхода. На приведенной структуре программы обозначены прямоугольниками, а данные – прямоугольниками со скругленными углами. Часть программ является штатной. Для проведения собственно трансформаций предлагается использование языка Пролог. Основными программами, требующими разработки, являются трансляторы T1, T2 и T3, осуществляющие преобразование OWL- и Prolog-базированных форм представления онтологий, линковщик онтологий и интерпретатор правил трансформаций на языке Prolog.

Направлением дальнейших исследований является более детальная проработка подхода и его программная реализация.

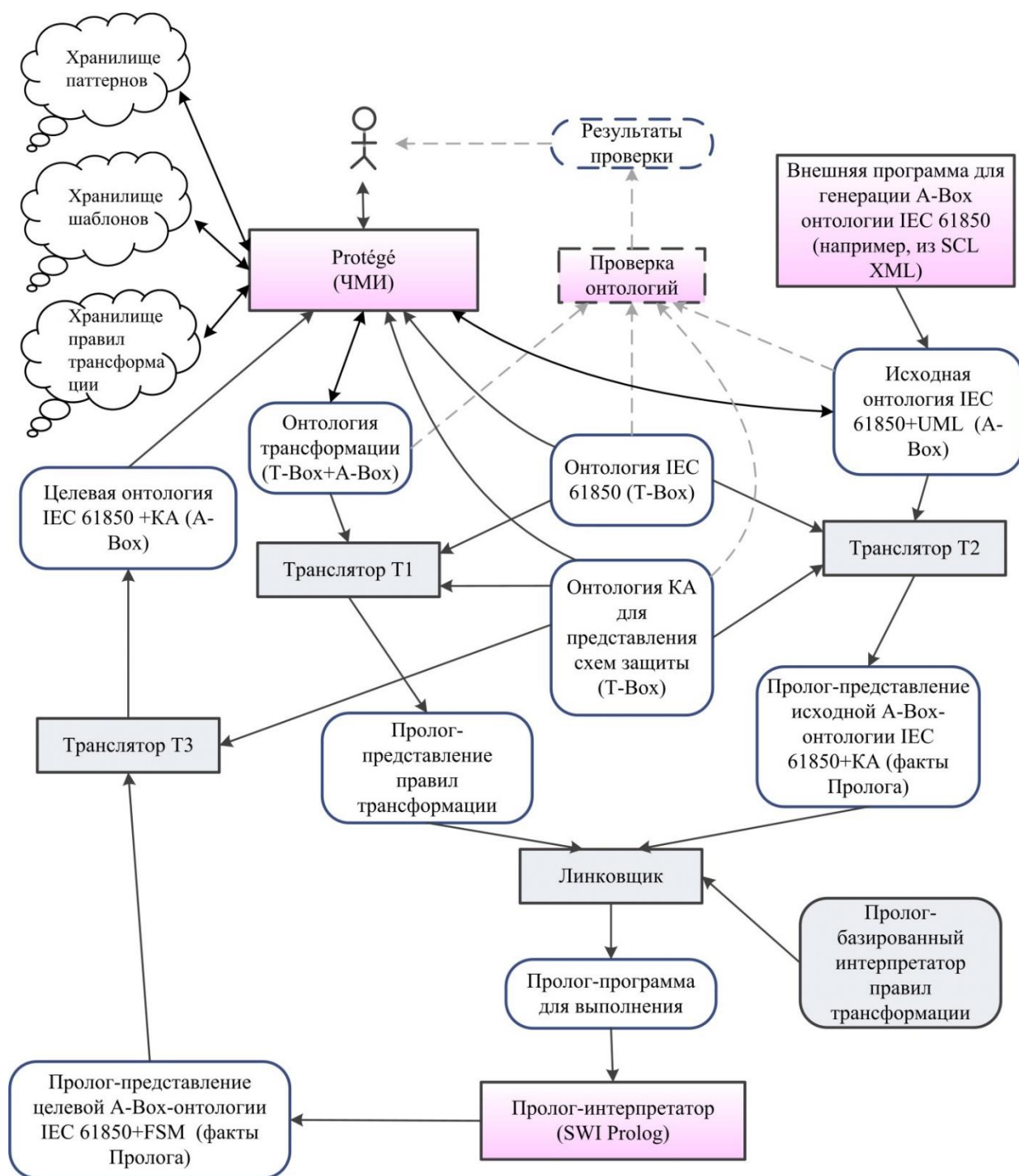


Рис. 8. Инструментарий для генерации схем защиты из описаний ЦПС IEC 61850

Список использованных источников

1. Васильев Д. А., Иващенко В. А., Лукьянов Д. В., Шабельникова А. Ю. Выбор структуры электрических сетей промышленных предприятий в условиях автоматизированного управления электропотреблением // Известия высших учебных заведений. Поволжский регион. Технические науки. – 2010. – № 2 (14). – С. 52–61.

2. Yang C.-W., Dubinin V., Vyatkin V. Ontology Driven Approach to Generate Distributed Automation Control from Substation Automation Design // IEEE Transactions on Industrial Informatics. – 2017, Volume 13, Issue 2. – pp. 668-679.

3. Войнов А.С., Дубинин В.Н., Сенокосов И.В., Климкина Л.П., Вяткин В.В. Функционально-блочное моделирование системы защиты от перегрузок цифровых подстанций МЭК 61850 // Междунар. науч.-техн. форум «Современные технологии в науке и образовании»: Сб. трудов. Том 4, Рязань, С. 160-165.

УДК 658.562.012.7

Д.А. Жуков, В.Н. Клячкин

**ИСПОЛЬЗОВАНИЕ АГРЕГИРОВАННЫХ КЛАССИФИКАТОРОВ
ПРИ МАШИННОМ ОБУЧЕНИИ В ЗАДАЧАХ
ТЕХНИЧЕСКОЙ ДИАГНОСТИКИ**

Ульяновский государственный технический университет
(Россия, Ульяновск, v_kl@mail.ru)

Аннотация. В задачах технической диагностики может использоваться бинарная классификация методами машинного обучения: по множеству показателей работы определяется, исправен объект или неисправен. Могут использоваться различные критерии качества классификации: процент ошибок при кросс-валидации, F-мера, учитывающая несбалансированность данных, и AUC – площадь под ROC-кривой. При выборе метода классификации применяется агрегирование базовых методов классификации тремя способами – по среднему значению, по медиане и по результатам голосования. Агрегированные методы во всех случаях дали лучшее значение критерия качества классификации, чем любой из отдельных базовых методов. В проведенном исследовании выявлено, что в состав агрегата чаще всего входят бэггинг деревьев решений и градиентный бустинг.

Ключевые слова: машинное обучение агрегированные классификаторы, бэггинг деревьев решений, градиентный бустинг.

D.A. Zhukov, V.N. Klyachkin

**AGGREGATED CLASSIFIERS IN MACHINE LEARNING IN THE
TASKS OF TECHNICAL DIAGNOSTICS**

Ulyanovsk State Technical University
(Russia, Ulyanovsk, v_kl@mail.ru)

Abstract. In the tasks of technical Diagnostics can be used a binary classification methods of machine learning: for a variety of performance determined defective or defective object. Different criteria may be used for quality classification: percentage of errors during cross-validation, F-measure, taking into