



OSTIS-2016

(Open Semantic Technologies for Intelligent Systems)

УДК 007:519.816

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ МОДЕЛИРОВАНИЯ СИСТЕМ НА ОСНОВЕ ТЕМПОРАЛЬНЫХ СЕТЕЙ ПЕТРИ

Еремеев А.П., Королев Ю.И.

*Национальный исследовательский университет «МЭИ»,
г. Москва, Россия*

Eremeev@appmat.ru

KorolevYu@gmail.com

В работе рассматривается вопрос разработки программного инструментария, предназначенного для моделирования сложных динамических систем и позволяющего оперировать временными (темпоральными) зависимостями. Моделирование осуществляется на основе формализма раскрашенных сетей Петри реального времени с поддержкой логики Аллена. Описывается прототип инструментария в среде *G2*, разработка программных средств на языках высокого уровня. Приведен пример использования созданного программного обеспечения в задаче разработки моделей объектов электроэнергетической системы.

Ключевые слова: интеллектуальная система поддержки принятия решений, темпоральные сети Петри, моделирование систем, разработка инструментария.

Введение

В настоящее время активно разрабатываются методы и программные средства проектирования математического и программного обеспечения перспективных интеллектуальных систем поддержки принятия решений (ИСППР), включая ИСППР реального времени (ИСППР РВ) семиотического типа, способных к развитию и адаптации к управляемому объекту и к изменениям внешней среды [Вагин и др., 2001]. Такие системы предназначены для помощи лицам, принимающим решения (ЛПР), при управлении и мониторинге сложных динамических систем (ДС) в условиях жестких временных ограничений и при наличии неопределенности в поступающей информации. ИСППР РВ относятся к классу динамических интеллектуальных систем (ДИС) [Осипов, 2008], основанных на использовании методов искусственного интеллекта и поиска решений на основе знаний. Одним из основных блоков ДИС является подсистема моделирования [Вагин и др., 2008], а потому разработка программных средств моделирования сложных ДС является важной задачей, возникающей при создании ИСППР РВ.

Учет времени необходим при решении задач диагностики и мониторинга сложного объекта, планирования действий, прогнозирования последствий решений, управления в реальном времени, обучения ЛПР. Поэтому при разработке

ИСППР РВ особое внимание должно уделяться средствам оперирования темпоральными зависимостями, как количественными, так и качественными. В ряде современных коммерческих инструментальных комплексах для построения сложных ДС (*G2*, *RTworks* и др.) имеются средства отображения времени и темпоральных зависимостей, однако они весьма примитивны и не соответствуют сегодняшним требованиям.

В основу системы моделирования современных ИСППР РВ необходимо заложить достаточно мощный формальный аппарат. Разработка абсолютно нового формализма сопряжена со значительными трудностями, поэтому эффективным путем является выбор существующего формализма и его модификация с целью разработки математического и программного обеспечения для моделирования и анализа сложных ДС, а также включения этих средств в состав ИСППР РВ. В качестве базового формализма для создания моделей, ориентированных на использование в составе ИСППР РВ семиотического типа, предлагается аппарат на основе раскрашенных сетей Петри (РСП), позволяющий оперировать темпоральной информацией [Еремеев и др., 2013].

В данной статье рассматриваются вопросы разработки и использования программного инструментария для моделирования сложных ДС (включая ДИС), в основе которого лежит формализм модифицированных РСП.

1. Базовый математический формализм

РСП [Jensen, 1992] – графоориентированный язык для проектирования, описания, имитации и контроля распределенных и параллельных систем. Графическими примитивами отображается течение процесса, а конструкциями специального языка имитируется необходимая обработка данных. Главным отличием РСП от классических сетей Петри (СП) является типизация данных, основанная на понятии множества цветов, которое аналогично типу в декларативных языках программирования. Класс РСП совпадает по выразительной мощности с классом СП, однако построенная на их базе модель получается, как правило, более компактной и структурированной.

Для практического использования РСП разработана система *CPN Tools*, которая поддерживает простой подкласс РСП, допускающий использование временных задержек на фишках. Подобный подход эффективен, если при моделировании предполагается соответствие каждой фишки некоторому ресурсу с определенными темпоральными ограничениями. Однако при создании моделей сложных ДС рационально использовать фишки как индикаторы состояния определенного объекта системы. Кроме того, простая поддержка временных меток в РСП делает крайне сложной задачу учета темпоральных зависимостей между объектами модели. Поэтому подход к обработке темпоральной информации, используемый в РСП, неэффективен при решении задачи моделирования сложных ДС.

В качестве эффективного средства моделирования и анализа процессов в сложных ДС (в том числе и в собственно ДИС) предлагается использовать аппарат раскрашенных сетей Петри реального времени с поддержкой темпоральной логики Аллена (РСП РВ ТЛА) [Еремеев и др., 2013]. Формализм РСП РВ ТЛА является расширением подкласса РСП РВ [Szpyrka, 2008] и обладает следующими ключевыми особенностями:

- каждому месту сети p ставится в соответствие временная метка $\tau(p)$. Фишки-токены места, временная метка которого положительна, недоступны для переходов сети;
- каждая дуга f имеет два параметра: весовое и временное выражения, причем результатом вычисления весового выражения $\varepsilon_\Sigma(f)$ является комплект фишек, результатом вычисления временного выражения $\varepsilon_T(f)$ является неотрицательное вещественное число;
- каждому переходу t ставится в соответствие некоторое логическое выражение типа $\{true, false\}$ – защитная функция $\gamma(t)$. Переход t может сработать только в том случае, если в результате вычисления выражения $\gamma(t)$ получается значение $true$.
- каждому переходу t ставится в соответствие приоритет $\pi(t)$, что позволяет напрямую моделировать детерминированный выбор.

При вычислении значений весовых и временных выражений дуг сети и защитных функций переходов используется функция подстановки β , которая замещает каждую переменную в выражениях определенным значением соответствующего типа. Функционирование модели на основе РСП РВ ТЛА осуществляется путем перехода сети из одного состояния в другое. Такой переход происходит вследствие одного из следующих событий:

- срабатывание перехода t в некоторой подстановке β ;
- уменьшение значений временных меток всех мест на фиксированную величину τ (временной шаг работы модели).

При этом безусловный приоритет имеет срабатывание перехода. Уменьшение значений временных меток позволяет только дожидаться момента, когда может сработать очередной переход.

Правила оперирования темпоральной информацией в РСП РВ ТЛА позволяют моделировать процессы функционирования ДС и оперировать как количественными, так и качественными темпоральными зависимостями. Для отображения количественных зависимостей используются конструкции временных меток мест и временных выражений дуг. Для работы с качественными зависимостями используются защитные функции переходов $\gamma(t)$, которые могут задаваться формулами темпоральной логики Аллена (ТЛА) [Allen, 1983] вида:

$$\varphi ::= \langle p_1 \rangle (\langle u_1 \rangle) \{ r_1, r_2, \dots, r_n \} \langle p_2 \rangle (\langle u_2 \rangle). \quad (1)$$

Обозначение $\langle p \rangle (\langle u \rangle)$ определяет временной интервал, на котором в месте сети p находится комплект фишек u . В каждый момент работы модели рассматриваются текущие, или последние завершившиеся интервалы. Формула ТЛА истинна (*true*), если между двумя интервалами соблюдаются отношения r_1, r_2, \dots, r_n , и ложна (*false*) в противном случае. Базисные отношения ТЛА приведены в [Allen, 1983].

Более подробные формальные определения рассматриваемого базового формализма РСП РВ ТЛА и описание принципов функционирования моделей на его основе приведены, в частности, в [Еремеев и др., 2015a]. Вопросы анализа и верификации таких моделей рассмотрены в работе [Еремеев и др., 2015b].

Преимуществом использования средств ТЛА является возможность использовать не только количественные временные меры, но и качественные отношения между интервалами, определяющими состояния отдельных объектов системы. Это позволяет, в частности, разбивать сети сложных моделей на более простые несвязные подсети и адекватно отражать неопределенность, присущую сложным ДС, что является одним из основополагающих принципов при проектировании перспективных ИСППР РВ и ДИС в целом.

2. Разработка инструментария

Компьютерное моделирование сложных ДС актуально как для теоретических исследований, так и для практического применения. Современные инструментальные средства конструирования ДИС, в частности, мощный инструментальный комплекс конструирования систем реального времени *G2* (*Gensym Corp.*, США) позволяют реализовывать подобные проекты.

2.1. Реализация прототипа в среде G2

G2 – это объектно-ориентированная интегрированная среда для разработки и сопровождения сложных приложений – интеллектуальных систем реального времени, использующих базы знаний. В отличие от систем, ориентированных на какую-то одну методологию или конкретную предметную область, *G2* интегрирует в себе множество взаимодополняющих методов искусственного интеллекта, что упрощает и ускоряет процесс разработки приложений и позволяет делать их достаточно универсальными. Программные продукты, разработанные с помощью *G2*, являются независимыми от аппаратного обеспечения, на котором они работают (т.е. являются переносимыми). Возможность простого манипулирования графическим представлением объектов в *G2* и составления схем, являющихся отображением технологических цепочек или алгоритмов обработки данных, обеспечивает базовые средства для построения проблемно-ориентированных языков визуального программирования. Реализованы версии *G2* для всех типов рабочих станций и персональных ЭВМ, а также обеспечивается возможность интеграции *G2* с большим числом программных и аппаратных средств самых разных фирм.

В системе *G2* был разработан прототип инструментария для моделирования процессов в сложных ДС на основе РСП РВ ТЛА. При этом был введен ряд допущений:

- в модели отсутствуют переменные, это приводит к тому, что любая подстановка перехода β является по сути тривиальной: $\beta = ()$;
- создание только безопасных сетей (не более одной фишки в каждом месте), весовыми выражениями дуг $\varepsilon_{\Sigma}(f)$ являются только единичные фишки;
- упрощенный алгоритм разрешения конфликта переходов;
- защитные функции переходов $\gamma(t)$ на основе ТЛА необходимо реализовывать в отдельных процедурах;
- временной шаг модели τ зафиксирован и равен 1 секунде.

В качестве примера рассмотрим модель системы экстренного торможения поезда, описанную в [Еремеев и др., 2013] (рисунок 1). В кабине машиниста каждые 60 секунд загорается световой сигнал, чтобы проверить, контролирует ли он

идущий поезд. Если машинист проигнорирует световой сигнал, то через 6 секунд включается звуковой сигнал. Затем, если машинист не деактивирует его в течение 3 секунд, срабатывает механизм аварийного торможения.

Формулы ТЛА применены в данном случае как защитные функции переходов *DisactLS* и *DisactSS*, обозначающих своевременную реакцию машиниста (ЛПР) на световой и звуковой сигнал соответственно:

- *Driver(active) {d} LightSig(on)* – переход *DisactLS* работает, когда фишка *active* в месте *Driver* появится в тот момент, когда в месте *LightSig* будет находиться фишка *on*;
- *Driver(active) {d} SoundSig(on)* – переход *DisactSS* работает, когда фишка *active* в месте *Driver* появится в тот момент, когда в месте *SoundSig* будет находиться фишка *on*.

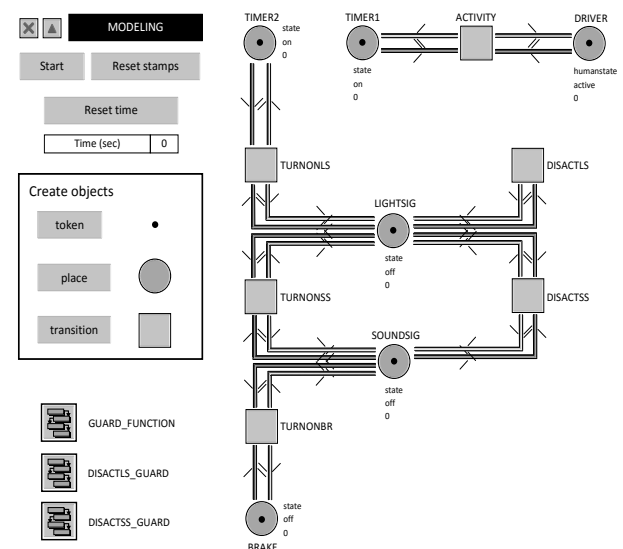


Рисунок 1 – Модель системы экстренного торможения поезда

Разработка прототипа инструментария в среде *G2* и создание на его базе моделей процессов позволили подтвердить целесообразность применения РСП РВ ТЛА для реализации моделей сложных ДС. Однако использование среды *G2*, несмотря на удобство разработки, существенно сужает область применимости полученного программного продукта. Поэтому было принято решение разработать расширенную версию инструментария на языках высокого уровня в среде *Microsoft Visual Studio (MVS)*.

2.2. Разработка на языках высокого уровня

Среда *MVS*, в частности, ориентирована на разработку приложений для платформы *.NET Framework*, основой которой является общезыковая среда исполнения *Common Language Runtime*, подходящая для разных языков программирования. Функциональные возможности этой среды доступны в любых языках программирования, использующих эту среду. Платформа *.NET* рассчитана на работу под операционными системами семейства *Microsoft Windows*, однако

существуют независимые проекты, позволяющие запускать программы *.NET* на других операционных системах (в частности, семейства *UNIX*).

Другой особенностью *MVS* является возможность использования в одном проекте нескольких языков программирования. Поддержка современных объектно-ориентированных языков высокого уровня, в частности, *C++* и *C Sharp*, делает эту среду перспективным инструментом для разработки наукоемких приложений.

Задачу разработки программы-инструментария можно разбить на две подзадачи:

- разработка функциональной логики работы приложения, «модели», содержащую всю наукоемкую часть задачи;
- разработка графического интерфейса пользователя (*graphical user interface, GUI*) – «представления», с помощью которого разработчик создает и настраивает «модель».

Для разработки *GUI* было принято решение использовать наиболее современную графическую *.NET* систему *Windows Presentation Foundation (WPF)* для построения клиентских приложений *Windows*. В основе *WPF* лежит векторная система визуализации, не зависящая от разрешения устройства вывода и созданная с учётом возможностей современного графического оборудования. Графической технологией, лежащей в основе *WPF*, является *DirectX*, в отличие от *Windows Forms*, где используется *GDI/GDI+*. Производительность *WPF* выше, чем у *GDI+* за счёт использования аппаратного ускорения графики через *DirectX*.

Применение *WPF* повлекло за собой решение использовать шаблон *Model View ViewModel, (MVVM)* – «модель»-«представление»-«модель представления» [Gossman, 2005]. «Модель представления», с одной стороны, является абстракцией «представления», а с другой – предоставляет данные из «модели». Классический шаблон «модель»-«представление»-«контроллер» (*Model View Controller*), при использовании которого изменения в пользовательском интерфейсе не влияют непосредственно на модель, а предварительно проходят через контроллер, в данном случае неудобен, поскольку в технологии *WPF* широко используется концепция «связывания данных», позволяющая связывать данные с визуальными элементами интерфейса в обе стороны.

В рамках предложенного подхода было разработано 3 группы элементов:

- элементы графического интерфейса (пространство имен *RTCPNtoolkit.View*);
- элементы логики функционирования (*RTCPNtoolkit.Model*);
- элементы, обеспечивающие взаимодействие (*RTCPNtoolkit.ViewModel*).

При этом первые две группы независимы друг от друга, а организация взаимодействия ложится на элементы третьей группы. Связь пространств имен *RTCPNtoolkit.View* и *RTCPNtoolkit.ViewModel* осуществляется преимущественно за счёт применения механизмов связывания данных (*data binding*), связь пространств имен *RTCPNtoolkit.Model* и *RTCPNtoolkit.ViewModel* осуществляется за счёт применения механизмов создания и отслеживания событий (*event-driven programming*). Таким образом, использование шаблона *MVVM* позволило полностью отделить задачу разработки функционирования моделей на основе РСП РВ ТЛА от задачи разработки *GUI*.

Тестирование разработанных инструментальных средств было проведено с использованием комплекса заранее подготовленных примеров. Большая часть этих примеров представляет собой искусственные конструкции, позволяющие проверить корректность разработанных алгоритмов функционирования моделей на основе РСП РВ ТЛА, в частности, алгоритмов определения допустимости перехода и разрешения переходов. Проведенное функциональное тестирование позволило усовершенствовать ряд шагов алгоритмов, повысив производительность программного продукта. На разработанный программный комплекс получено свидетельство о государственной регистрации программы для ЭВМ № 2015616435 «Инструментарий для разработки моделей систем на основе темпоральных сетей Петри с поддержкой логики Аллена».

3. Применение инструментария

Предложенный формальный аппарат РСП РВ ТЛА и соответствующий программный инструментарий на его основе был применен в рамках НИР по модернизации электродинамической модели МЭИ (ЭДМ МЭИ), проводимых в плане приоритетных исследований по «интеллектуальной энергетике» с созданием комплекса соответствующих ИСПП РВ для мониторинга, диагностики и управления энергообъектами.

ЭДМ МЭИ позволяет адекватно воспроизводить различные нормальные и аварийные режимы электроэнергетической системы (ЭЭС) и предназначена для исследования поведения новых управляемых элементов ЭЭС, для испытания натуральных устройств автоматического регулирования, противоаварийного управления, релейных защит в условиях, максимально приближенных к реальным, а также проверки и доработки алгоритмов для микропроцессорных регуляторов и устройств [Шаров и др., 2005]. Однако устаревание оборудования и аппаратных компонентов создает сложности в обслуживании и настройке алгоритмов управления. Поэтому в настоящее время проводятся масштабные работы по модернизации оборудования и, в частности, замене аналоговых устройств программируемыми контроллерами.

В рамках проводимых исследований особое внимание уделяется разработке моделей объектов ЭЭС. Разработка осуществляется с применением современных контроллеров *CompactRIO* – многофункциональной реконфигурируемой встраиваемой системой сбора данных и управления. При программировании контроллеров применяется *LabVIEW*, графическая среда разработки и платформа для выполнения программ. В отличие от *SCADA*-систем, которые ориентированы на разработку автоматизированных систем управления технологическим процессом, среда *LabVIEW* в большей степени ориентирована на решение задач в области автоматизированных систем научных исследований.

Программа *LabVIEW* является виртуальным прибором (ВП). В процессе моделирования объекта ЭЭС разрабатываются модели отдельных составляющих элементов этого объекта, каждая из которых является проектом (группой ВП) в среде *LabVIEW*. Архитектура таких приложений соответствует архитектуре контроллеров *CompactRIO* и представлена на рисунке 2. Каждое приложение использует три процессора:

- процессор ПК под управлением *Windows*;
- процессор блока *CompactRIO* под управлением ОС жесткого реального времени;
- программируемую логическую интегральную схему (ПЛИС).

Рациональным подходом является разбиение общей задачи моделирования на отдельные подзадачи для каждого процессора, в соответствии с функциональными характеристиками этих подзадач и техническими характеристиками процессоров. Обработка поступающих (или моделируемых) сигналов осуществляется на уровне ПЛИС, контроллер реального времени обрабатывает задачи, связанные с управлением ВП нижнего уровня. ВП на *Windows* используется для задания параметров моделирования и управления всем приложением.

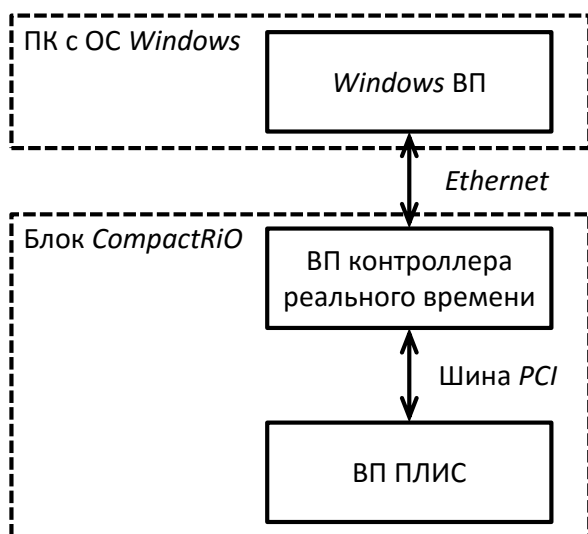


Рисунок 2 – Архитектура приложения *LabVIEW* для моделирования отдельного элемента объекта ЭЭС

Такая архитектура позволяет отдельно разрабатывать каждый элемент моделируемого объекта с использованием современных методов и подходов из области ЭЭС. Однако при этом возникает задача объединения таких подмоделей в единую модель объекта сети, для решения которой был применен подход на основе РСП РВ ТЛА. Архитектура разработанного приложения приведена на рисунке 3.

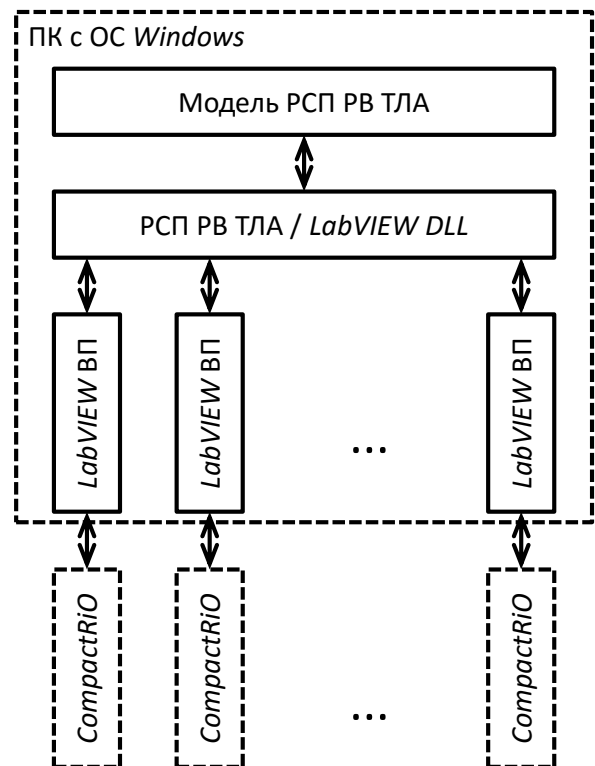


Рисунок 3 – Архитектура приложения для моделирования объекта ЭЭС

Проблема организации взаимодействия между ВП верхнего уровня и созданным инструментарием решается с помощью библиотеки *DLL* (*Dynamic Link Library*) – промежуточной библиотеки, позволяющей осуществлять двухстороннюю связь. При этом *DLL*-библиотека поддерживает работу с разделяемой памятью (*shared memory*), что позволяет увеличить скорость прохождения данных между отдельными элементами приложения. Данный подход предпочтителен, так как в его основе лежит модульный принцип: можно легко заменить одну *LabVIEW*-модель элемента объекта другой (при этом необходимо контролировать только интерфейсную часть ВП, который взаимодействует с промежуточной библиотекой). Поэтому предложенная архитектура удобна при проведении научных исследований в области ЭЭС.

Архитектура приложения позволяет использовать разработанную связующую сеть РСП РВ ТЛА в двух режимах работы:

- в режиме тестирования и отладки, когда сеть РСП РВ ТЛА рассматривается отдельно от *LabVIEW*-моделей, параметры функционирования

которых задаются с помощью вспомогательных подсетей. Использование формул ТЛА позволяет разрабатывать модели таким образом, чтобы эти подсети были не связаны напрямую (дугами) с основной сетью модели;

- в режиме моделирования, когда связующая сеть РСП РВ ТЛА получает данные непосредственно из *LabVIEW*-моделей. При этом модель фактически включается в контур управления и реализуется принцип модельно-управляемой разработки (*model-driven development*).

Предложенная архитектура применима не только для решения задачи моделирования объектов ЭЭС, но и в других приложениях, в том числе и в составе ДИС, а также при разработке их подсистем моделирования. Эксперименты (испытания) показали эффективность подхода к разработке моделей сложных ДС на основе аппарата РСП РВ ТЛА на примере моделирования функционирования объектов ЭЭС для использования в составе ДИС типа ИСППР РВ.

Заключение

В статье затронут вопрос разработки программного обеспечения для создания моделей сложных ДС на основе модификации РСП. Дано краткое описание базового формализма, приведено описание прототипа инструментария для моделирования на основе РСП РВ ТЛА, разработанного в среде *G2*. Функционирование программного приложения проиллюстрировано на примере. Отмечены недостатки комплекса *G2* – высокая стоимость комплекса и существенное сужение области применимости разрабатываемого приложения.

Рассмотрена полная и достаточно универсальная в плане как применения в составе ДИС типа ИСППР РВ семиотического типа, так и автономного использования, версия программного продукта в среде *Microsoft Visual Studio* на языке *C Sharp*. Предложена архитектура разработанного инструментария на основе шаблона *Model-View-ViewModel*. Приведен пример использования формализма РСП РВ ТЛА и инструментария моделирования на его основе в рамках работ по разработке моделей объектов ЭЭС. Предложена модульная архитектура приложения, показано, что модели на основе РСП РВ ТЛА удобно использовать при проведении научных исследований в области ЭЭС. Сделан вывод о том, что подход, при котором сеть РСП РВ ТЛА используется для объединения и организации взаимодействия отдельных модулей, можно использовать и в других задачах как автономно, так и в составе ДИС типа ИСППР РВ.

Работа выполнена при поддержке Министерства образования и науки Российской Федерации (государственное задание № 2.737.2014/К), РФФИ (проекты №14-01-00427, 15-07-04574, 16-37-00309) и Фонда содействия инновациям.

Библиографический список

- [Вагин и др., 2001] Вагин В.Н., Еремеев А.П. Некоторые базовые принципы построения интеллектуальных систем поддержки принятия решений реального времени // Известия РАН. Теория и системы управления, № 6, 2001. – С. 114-123.
- [Вагин и др., 2008] Вагин В.Н., Еремеев А.П. Исследования и разработки кафедры прикладной математики по конструированию интеллектуальных систем поддержки принятия решений на основе нетрадиционных логик // Вестник МЭИ, № 5, 2008. – С. 16-26.
- [Еремеев и др., 2013] Еремеев А.П., Королев Ю.И. Реализация интеллектуальных систем реального времени на основе сетей Петри с поддержкой темпоральных зависимостей // Программные продукты и системы, № 3 (103), 2013. – С. 88-94.
- [Еремеев и др., 2015a] Еремеев А.П., Королев Ю.И., Кутепов В.П., Фальк В.Н. Формализм сетей Петри с поддержкой темпоральных зависимостей // Труды Конгресса по интеллектуальным системам и информационным технологиям "IS&IT'15". Научное издание в 3-х томах. – Таганрог: Изд-во ЮФУ, 2015. – Т. 2. – С. 118-131.
- [Еремеев и др., 2015b] Еремеев А.П., Королев Ю.И. Анализ и верификация моделей процессов в сложных динамических системах // Искусственный интеллект и принятие решений. – 2015. – № 1. – С. 45-56.
- [Осипов, 2008] Осипов Г.С. Динамические интеллектуальные системы // Искусственный интеллект и принятие решений, № 1, 2008. – С. 47-54.
- [Шаров и др., 2005] Шаров Ю.В., Бейм Р.С., Карымов Р.Р., Сыромятников С.Ю. Электродинамическая модель электроэнергетических систем // Электрические станции, № 5, 2005. – С. 58-63.
- [Allen, 1983] Allen J.F. Maintaining knowledge about temporal intervals // Communications of the ACM. – 1983. – Vol. 26. – No. 11. – P. 832-843.
- [Gossman, 2005] Gossman J. Introduction to Model/View/ViewModel pattern for building WPF apps [Электронный ресурс] // MSDN Blogs. [сайт]. [2005]. URL: <http://blogs.msdn.com/b/johngossman/archive/2005/10/08/478683.aspx> (дата обращения: 07.04.2015).
- [Jensen, 1992] Jensen K. Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use. – Vol 1-3. – Springer-Verlag, 1992-1997.
- [Szpyrka, 2008] Szpyrka M. Modelling and Analysis of Real-Time Systems with RTCP-Nets // Petri Net, Theory and Applications. I-Tech Education and Publishing. - 2008. – P. 17-40.

DEVELOPMENT OF THE SOFTWARE FOR SYSTEMS MODELLING BASED ON TEMPORAL PETRI NETS

Eremeev A.P., Korolev Y.I.

National Research University «Moscow Power Engineering Institute», Moscow, Russia

Eremeev@appmat.ru

KorolevYu@gmail.com

In the paper the software development of the toolkit for complex dynamic system modelling that allows to operate with temporal dependencies is described. Modelling is based on the formalism of real-time coloured Petri nets supporting Allen's logic. The toolkite prototype designed in *G2* IDE and the development of software in high level languages are considered. The example of using the designed software to model objects of electric power system is given. The possibility of using the designed software in intelligent decision support systems is mentioned.

Keywords: intelligent decision support system, temporal Petri nets, system modelling, toolkit design.