

Об одном подходе к синтезу моделей динамических цифровых двойников

А.И. Водяхо¹, Н. А. Жукова², С. А. Аббас³,
И. А. Куликов⁴, Ф. А. Анаам⁵

Санкт-Петербургский государственный электротехнический университет
«ЛЭТИ» им. В.И. Ульянова (Ленина)

¹aivodyaho@mail.ru, ²nazhukova@mail.ru, ³saddamabbas077@gmail.com,

⁴i.a.kulikov@gmail.com, ⁵fahemye@gmail.com

Аннотация. Модельный подход находит все более широкое применение в современных кибер-физических системах. В частности, все более широкое практическое применение находят цифровые двойники. В докладе рассматривается один из возможных подходов к построению цифровых двойников. Отличительной особенностью развиваемого подхода является использование структурно-функциональной модели, которая синтезируется и поддерживается в актуальном состоянии автоматически. В докладе рассматриваются алгоритмы синтеза подобных моделей.

Ключевые слова: модельный подход; кибер-физические системы; цифровые двойники; синтез моделей

I. ВВЕДЕНИЕ

Характерной особенностью современного этапа развития общества является переход к цифровой экономике, основанной на знаниях. Становление и развитие информационного общества связано, в первую очередь, с построением и поддержанием в актуальном состоянии глобальных информационных пространств, в состав которых, помимо собственно информационных ресурсов, могут быть интегрированы сущности разной физической природы. Примером могут служить кибер-физические системы (КФС), социо-кибернетические системы, природные системы и т. п. Существенная часть антропогенных систем, в частности, КФС являются элементами других систем, т. е. являются элементами системы систем (System of Systems, SoS).

Модельный подход [1] в настоящее время находит самое широкое применение при построении современных КФС [2] для решения задачи интеграции гетерогенных систем. При построении крупномасштабных гетерогенных информационных систем (ИС) в настоящее время активно используются две взаимосвязанных парадигмы: цифровые двойники (Digital Twins, DT) и цифровые нити (Digital Threads, DTh).

Понятие DT обычно определяют как «виртуальное представление объекта любой природы, которое двунаправленно связано с двойником» [3].

DT можно рассматривать как модель, содержащую определенные знания, необходимые для управления данным элементом системы.

DTh обычно определяют, как «набор моделей для поддержки всех заинтересованных сторон в течение всего жизненного цикла системы или объекта» [4].

Отличительной особенностью современных КФС, особенно построенных по принципу SoS, является то, что их структура и поведение может изменяться во времени. Другими словами, модель, которая ставится в соответствие некоторому элементу системы, должна быть динамической.

Можно определить следующие основные задачи, решение которых должен обеспечивать DT:

- получение значений отдельных параметров, характеризующих текущую структуру и поведение системы;
- автоматическое построение структурно-функциональной модели основной (наблюдаемой) системы;
- подтверждение корректности (адекватности модели);
- корректировка модели на основе информации о внешних и внутренних событиях;
- приведение структуры наблюдаемой системы в соответствие с моделью.

При этом можно выделить следующие подходы к использованию парадигмы DT.

- DT рассматривается как средство построения SOS [5], т. е. как механизм интеграции гетерогенных компонентов. В этом случае каждая система, являющаяся частью SOS, представляется в системе верхнего уровня своим DT.
- DT является полным функциональным аналогом наблюдаемой и управляемой системы (НУС) и может ее замещать, например, на этапе настройки, моделирования отдельных ситуаций или в тренажерах.
- DT рассматривается как порт доступа к НУС, в этом случае можно говорить, что он используется в режиме контроллера. Если структура и поведение НУС остается неизменной, то это обычный контроллер, однако если структура и поведение НУС изменяются во времени, то DT должен отслеживать эти изменения.
- DT рассматривается как средство реализации концепции DTh, DT в этом случае рассматривается как набор моделей,

принадлежащих к разным стадиям жизненного цикла НС.

II. ПРЕДЛАГАЕМЫЙ ПОДХОД

Для реализации описанной выше функциональности предлагается следующий подход.

- Модель наблюдаемой системы управляется потоком событий, данные о которых содержатся в лог-файлах.
- В общем случае модель описывается в виде мультиграфа, представляющего собой суперпозицию 4 графов: графа потока управления, графа потока данных, графа потока запросов и ресурсного графа. Ресурсный граф описывает текущую структуру наблюдаемой системы. Данный граф рассматривается как метамодель. Исходный граф строится на основе априорных знаний о структуре и поведении наблюдаемой системы. В процессе функционирования граф изменяется.
- Для описания структуры и поведения НУС системы предлагается использовать многоуровневые относительно конечные автоматы [5].

Типовая структура многоуровневой системы, использующей DT, показана на рис. 1.

Кроме собственно многоуровневого НУС (МНУС) система содержит DT, с которым она связана внутренним интерфейсом. DT включает два основных элемента: модель НУС и процессор модели (ПМ), все обращения к НУС направляются к DT и обрабатываются с использованием модели. За поддержание модели в актуальном состоянии отвечает отдельный процесс, реализуемый ПМ.

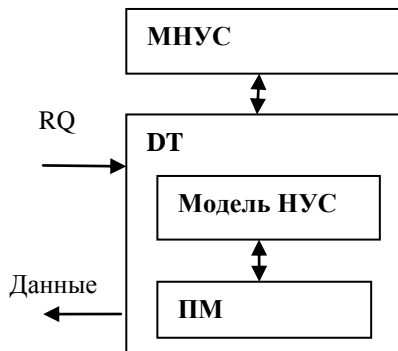


Рис. 1. Обобщенная структура системы, использующей DT

В самом общем виде процесс взаимодействия DT и НУС реализуется следующим образом. Все обращения к НУС (RQ) выполняются через DT и обрабатываются ПМ, который проверяет актуальность модели и выполняет запрос к модели. Если требуется получить данные о состоянии НУС, то осуществляется обращение к модели в режиме чтения. Если запрос содержит требование изменить состояние НУС, то изменяется сама модель и запускается процедура приведения в соответствие состояния НУС с моделью.

На практике для управления реальными системами используются многоуровневые модели. Если сложность системы невысокая, то обычно имеется возможность поддерживать всю многоуровневую модель в актуальном состоянии. Реально это не всегда возможно. Тогда в актуальном состоянии поддерживается только модель верхнего уровня, а модели, соответствующие нижним уровням, достраиваются при получении запроса.

III. ЗАДАЧА СИНТЕЗА МОДЕЛИ.

Для построения модели может использоваться процедура многоуровневого индуктивного синтеза, в соответствии с которой синтез осуществляется последовательно в направлении от нижнего уровня к верхнему. В самом общем виде процедура синтеза реализуется следующим образом.

1. Определяются элементы, относящиеся к i -му уровню.
2. Проверяется наличие связей между элементами, относящимися к i -му уровню.
3. Выполняется перекодирование группы элементов, относящихся к i -му уровню, при этом каждая группа элементов рассматривается как отдельный новый элемент.
4. Определяются характеристики новых элементов в терминах $(i+1)$ -го уровня и строится новое признаковое пространство.
5. Процедура повторяется для вышележащих уровней.

Следует заметить, что на практике редко требуется использовать полный граф. Часто оказывается достаточным отслеживать только потоки управления или только изменения структуры НУС. Можно выделить следующие наиболее часто встречающиеся постановки задач синтеза моделей, входящих в состав DT:

- предметом наблюдения и управления является только многоуровневая структура;
- предметом наблюдения и управления являются бизнес процессы (потоки управления, потоки данных или потоки запросов);
- требуется управлять как структурой, так и поведением НУС.

Для решения задач автоматического построения (синтеза) моделей могут быть использованы разные подходы. В простейших случаях, когда требуется построить топологию сетевой структуры, может оказаться достаточным использовать протокол SNMP [6]. В более сложных случаях, когда требуется построить структурную модель многоуровневой системы можно использовать алгоритмы синтеза многоуровневых относительно-конечных автоматов [5].

Если речи идет о синтезе моделей поведения (синтезе бизнес-процессов), то естественным является использование алгоритмов Process Mining [7]. В общем случае, возможно совместное использование перечисленных выше подходов.

IV. ПРАКТИЧЕСКОЕ ПРИМЕНЕНИЕ ПРЕДЛАГАЕМОГО ПОДХОДА

Предлагаемый подход был использован авторами при создании ряда реальных систем.

Впервые данный подход был применен при построении системы мониторинга технического состояния сети кабельного телевидения [8]. Результаты применения разработанной системы показали, что предложенный подход возможно использовать для автоматического построения динамических структурных моделей систем, что позволяет быстро удаленно решать проблемы обнаружения неисправностей. Основным преимуществом разработанного подхода является снижение совокупной стоимости владения системой [9].

Вторым объектом внедрения была производственная система управления крановым оборудованием, которая показала возможность использования предлагаемого подхода для решения задач сбора данных в гетерогенных системах, построенных на платформе промышленного интернета вещей. Использование предложенного подхода позволило сократить задержки в получении данных и построить фрагмент корпоративного графа знаний [10].

Позже была разработана система автоматического построения учебных планов. По сути, это задача автоматического построения бизнес-процессов. Этот пример иллюстрирует возможность использования предлагаемого подхода не только для кибер-физических, но и социально-кибернетических систем.

V. ЗАКЛЮЧЕНИЕ

В результате проведенных исследований был разработан подход к автоматическому построению моделей систем, который может быть использован при построении ДТ сложных систем разной природы с динамической структурой и поведением.

Дальнейшие исследования по данной тематике авторы планируют в двух направлениях: в направлении расширения сферы применения развиваемого подхода и в направлении его использования на разных этапах жизненного цикла кибернетических систем и социо-кибернетических систем. В последнем случае речь идет об автоматическом построении ДТ моделей на базе моделей, используемых на этапе проектирования, в частности UML и SysML моделей.

СПИСОК ЛИТЕРАТУРЫ

- [1] Weikens T., Lamm J., Roth S., Walker M. *Model-Based System Architecture*; JohnWiley & Sons, Inc.: Hoboken, NJ, USA, 2016; 375 p.
- [2] Patnaik S. *New Paradigm of Industry 4.0 Internet of Things, Big Data & Cyber Physical Systems*, Springer Nature: Cham, Switzerland, 2020, p. 187.
- [1] ISO/IEC JTC 1/SC 41/WG 6 (Digital Twin). Available online: https://www.iec.ch/ords/f?p=103:14:708632010937904:FSP_ORG_ID:27186 (accessed on 15 October 2021).
- [3] Lawrence S. Gould. *What Are Digital Twins and Digital Threads?* Available online: <https://www.gardnerweb.com/articles/what-are-digital-twins-and-digital-threads> (accessed on 15 October 2021).
- [4] Jamshidi M. *Systems of systems engineering. Principles and applications*. 2009 by Taylor & Francis Group, LLC CRC Press is an imprint of Taylor & Francis Group, Boca Raton, FL 480 p.
- [5] Tianxing M., Osipov V., Vodyaho A.I., Lebedev S., Zhukova N. *Distributed Technical Object Model Synthesis Based on Monitoring Data*. *Int. J. Knowl. Syst. Sci.* 2019, 10, 27–43.
- [6] Mauro D., Schmidt K. *Essential SNMP*, 2nd ed.; O'Reilly Media, Inc.: Newton, MA, USA, 2005, 426p.
- [7] Van der Aalst, W. *Process Mining Data Science in Action*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2016; 468p.
- [8] Osipov V., Vodyaho A., Zhukova N. *About One Approach to Multilevel Behavioral Program Synthesis for Television Devices*. *Int. J. Comput. Commun.* 2017, 11, 17–25.
- [9] Vodyaho A., Osipov V., Zhukova N., and Chernokulsky V. "Data collection technology for ambient intelligence systems in internet of things," *Electronics*, vol. 9, no. 11, p. 1846.