

**Синтез модульных конечно-автоматных моделей по библиотеке модулей
с помощью SAT-решателей**

Чухарев К.И., Университет ИТМО

Научный руководитель – к.т.н. Чивилихин Д.С., Университет ИТМО

В работе предлагается метод для построения минимальной модульной системы управления по примерам поведения и заданной библиотеке модулей, основанный на применении SAT-решателей. Рассматривается как задача синтеза логической схемы без состояния, так и задача синтеза системы, содержащей модули, представляемые в виде конечных автоматов.

Введение. Конечно-автоматные модели широко используются в программной инженерии, особенно при разработке, проектировании и отладке киберфизических систем (КФС), большинство из которых являются по своей природе модульными и распределенными. Создание модульных и распределенных алгоритмов для КФС – сложная задача, в большинстве случаев выполняемая инженерами вручную, что требует существенных ресурсов. Альтернативой является *автоматизированный синтез* управляющих алгоритмов контроллеров с помощью методов машинного обучения.

Целью настоящей работы является разработка методов синтеза конечно-автоматных моделей модульных логических контроллеров по примерам поведения и заданной *библиотеке модулей*.

Основная часть. В данной работе для автоматизированного синтеза конечно-автоматных моделей используются методы программирования в ограничениях, а именно, исходная задача синтеза сводится к задаче выполнимости булевой формулы (*Boolean Satisfiability* – SAT), для непосредственного решения которой используются сторонние программные средства – SAT-решатели. Выбранный подход является наиболее эффективным среди аналогов, а также гарантирует точное соответствие заданной спецификации и минимальность получаемого решения – конечно-автоматной модели.

Основой сведения к SAT является метод синтеза модульных моделей с произвольной композицией модулей по примерам поведения, ранее разработанный автором настоящей работы. В указанном методе синтезируемая модульная система состоит из заданного числа модулей, представляемых конечными автоматами с заданными параметрами, объединенных в сеть произвольным образом, образуя тем самым композитный функциональный блок. Отличительной особенностью разработанного в данной работе метода является то, что каждый модуль синтезируемой модульной системы берется из заданной библиотеки модулей, то есть имеет заранее заданное поведение (необязательно описываемое в виде конечного автомата). Такой подход позволяет синтезировать систему, например, из доступных блоков стандартной библиотеки модулей используемой реализации стандарта IEC 61499.

Для апробации прототипа реализации разработанного метода была рассмотрена задача синтеза минимальной логической схемы, реализующей заданную логическую функцию, используя заданный набор логических вентилях (например, И, ИЛИ, НЕ и т. п.). Можно заметить, что, например, блок «НЕ» фактически использует только один вход – второй должен оставаться «неподключенным». Это достигается объявлением дополнительных ограничений, однако в общем случае все модули обладают двумя входами и одним выходом, так как заранее неизвестно, какие именно из блоков будут иметь тип «НЕ» – определение этого и является сутью рассматриваемой задачи. Экспериментальное исследование подтвердило работоспособность разработанного метода, но также и предсказуемую низкую масштабируемость – некоторые логические функции (даже от небольшого числа переменных) выражаются с использованием заданных вентилях только в виде формул

большого размера, что затрудняет типичный процесс нахождения минимальной формулы – перебор «снизу вверх». Однако было также выявлено, что время синтеза логических схем размера больше минимального слабо зависит от непосредственного размера (в контексте задачи – числа модулей системы), что открывает возможности по эффективному использованию перебора «сверху вниз» для нахождения минимальных формул.

Далее была рассмотрена задача синтеза модульной системы, содержащей модули, представимые в виде конечных автоматов, что потребовало модификации сведения для учета состояний, в которых находятся эти конечные автоматы в процессе работы всей системы. Все блоки модульной системы в общем случае имеют интерфейс (входы и выходы) размера максимального блока из заданной библиотеки модулей, включая размер автомата. Аналогично ситуации с вентилем «НЕ», в зависимости от фактического типа блока (динамически определяемого SAT-решателем в ходе его работы), некоторых входы, выходы и состояния отмечаются как «неиспользуемые». Это приводит к тому, что максимальный по размеру блок в заданной библиотеке оказывает сильное влияние на эффективность разработанного метода.

Результаты и выводы. Разработан метод синтеза модульной системы по примерам поведения с использованием модулей из заданной библиотеки модулей, основанный на применении SAT-решателей. Была рассмотрена как вариация задачи синтеза системы без состояния, так и задача синтеза системы с модулями, представляемыми в виде конечных автоматов. Экспериментальное исследование подтвердило сильную зависимость эффективности метода от содержания заданной библиотеки, а именно, от размера максимального блока в ней.

Разработанные методы реализованы на языке программирования Kotlin в программном средстве fbSAT с открытым исходным кодом (<https://github.com/ctlab/fbSAT/tree/dev>).

Автор

Чухарев К.И.

Научный руководитель

Чивилихин Д.С.
