

Аргиров Виктор Сергеевич

ТРАНСЛЯЦИЯ SDL-СПЕЦИФИКАЦИЙ В МОДИФИЦИРОВАННЫЕ РАСКРАШЕННЫЕ СЕТИ ПЕТРИ

1. ВВЕДЕНИЕ

Верификация распределенных систем, таких как коммуникационные протоколы, — актуальная проблема современного программирования. Для представления распределенных систем часто используется язык выполнимых спецификаций SDL [1, 2], принятый в качестве стандарта ITU. Преимущество SDL в его выразительной силе однако, именно оно и затрудняет анализ и верификацию спецификаций этих систем. Один из подходов состоит в автоматическом переводе спецификаций распределенных систем в модели, для которых разработаны эффективные методы анализа. В качестве моделей выбраны модифицированные раскрашенные сети Петри, названные иерархическими временными типизированными сетями (ИВТ-сетями) [3]. ИВТ-сети расширяют безопасные раскрашенные сети Петри посредством понятий времени (семантика Мерлина), приоритетов, а также специальных мест, представляющих очереди фишек.

В данной работе описывается автоматический перевод статического подмножества SDL и генерация сетевой модели в системе SPV (SDL Protocol Verifier). Эта работа частично поддержана грантом РФФИ № 03-07-90331В.

2. СИСТЕМА SPV

Система SPV представляет собой интегрированный программный комплекс для проектирования, анализа и симуляции моделей ИВТ-сетей.

Комплекс SPV включает:

- графический редактор иерархических сетевых моделей;
- транслятор с языка SDL;
- симулятор — блок имитационного моделирования и отладки.

Типичный цикл работы пользователя в системе SPV выглядит следующим образом. Строится исходная сетевая модель исследуемой системы, что может быть сделано непосредственно в графическом редакторе системы

SPV, либо эта сетевая модель получается каким-либо другим способом, например, как результат трансляции с языка выполнимых спецификаций SDL. Затем проводится симуляция модели в ручном или автоматическом режиме. Система позволяет визуально контролировать ход симуляции, отслеживать изменение данных, устанавливать контрольные точки останова и т. п. По результатам симуляции исходная модель уточняется, и цикл разработки повторяется до получения удовлетворительных результатов.

Симулятор позволяет проследить за процессом функционирования сети при заданной начальной разметке. При этом часто удается обнаружить семантические ошибки модели, проявляющиеся в ненадлежащем поведении сети.

3. ТРАНСЛЯТОР ИЗ SDL В СЕТЕВЫЕ МОДЕЛИ

Алгоритм перевода SDL-спецификаций в сетевые модели системы SPV реализован методом двухпроходной трансляции. На первом проходе модулем синтаксического анализатора строится внутреннее представление спецификации, а на втором — по этому представлению генерируется иерархическая сетевая модель. Синтаксический анализатор языка SDL построен с помощью программного средства Bison.

Транслятор функционирует следующим образом. Модуль анализатора обрабатывает текстовый файл, содержащий SDL-спецификацию, т. е. осуществляет лексическую свертку и синтаксический разбор и строит внутреннее представление спецификации. В случае отсутствия ошибок запускается модуль генерации сетевой модели. Сначала строится внутреннее представление ИВТ-сети, а затем осуществляется ее визуализация в системе SPV. Связь между модулями является однонаправленной и реализуется посредством сервисных функций (без использования глобальных переменных).

Иерархическая сетевая модель (ИВТ-сеть) — это композиция множества неиерархических сетей, называемых страницами. Страницы могут содержать вершины специального типа, которые называются модулями и соединяются с местами на странице по тому же принципу, что и переходы. Модуль представляет подсеть, располагающуюся на отдельной странице, которая в свою очередь может содержать модули.

Такая страница называется подстраницей страницы, на которой располагается модуль. Подстраница содержит копии всех мест, с которыми связан модуль. Место-копия может быть входным местом для некоторого пе-

рехода или модуля на подстранице тогда и только тогда, когда его прототип является входным местом для модуля, представляющего подстраницу. Аналогично, только копия выходного места-прототипа может быть выходным местом некоторого перехода или модуля на подстранице.

Поведение иерархической сети определяется поведенчески эквивалентной ей неиерархической сетью, получающейся при замещении всех модулей страницами, которые они представляют. При этом каждый модуль вместе со своими дугами удаляется со страницы, а на его место помещается подсеть, располагавшаяся на подстранице. Соединение сетей происходит по местам: каждое место-прототип склеивается со всеми своими копиями.

Построение внутреннего представления сети производится по шагам, соответствующим этапам в описании алгоритма трансляции [4]. В первую очередь создается корневой уровень иерархической сети, который состоит из модулей системы SPV, представляющих блоки SDL-спецификации. Также на этом этапе создаются места-очереди, моделирующие каналы. Эти модули и места соединяются дугами в соответствии с описанием SDL-системы. Следующие четыре шага генерации выполняются последовательно для каждого блока.

На втором шаге генерируется сеть, реализующая блок SDL-спецификации. Эта сеть, в свою очередь, содержит модули, соответствующие процессам, и места-очереди, моделирующие маршруты сигналов. Эти модули и места соединяются дугами в соответствии с описанием блока.

На следующем шаге генерируется сеть, реализующая процесс SDL-спецификации. Она содержит модули, соответствующие SDL-переходам процесса, места, моделирующие переменные и таймеры, и некоторые служебные места. На этом этапе построения сети дуги не создаются, а достраиваются на следующем шаге, поскольку нельзя заранее сказать, в каких переходах используется переменная или таймер.

На следующем шаге — трансляции SDL-переходов — создается подсеть, в которой реализуется логика SDL-перехода. В процессе построения этой сети создаются дуги для сети третьего уровня.

На завершающем шаге создаются модули, реализующие процедуры и действия с таймерами, если таковые есть.

При построении сети в системе SPV используются средства создания иерархических сетевых моделей, предоставляемые этой системой. После того как сеть создана, осуществляется размещение ее элементов на плоскости, т. е. каждому элементу приписываются координаты на соответствующих страницах системы SPV. При этом фрагменты сети, реализующие действия SDL-переходов, размещаются типовым образом.

4. ЭКСПЕРИМЕНТ С ПРОТОКОЛОМ INRES

Описание протокола InRes приведено в работе [5] в составе системы InRes на языке SDL, а также в работе [6].

Полученная при трансляции сеть имеет пять уровней вложенности. Сеть первого уровня содержит модули, моделирующие блоки SDL-спецификации. Сети второго уровня вложенности реализуют процессы, а третьего — SDL-переходы. Сети четвертого уровня представляют действия SDL-переходов. Некоторые из этих сетей также содержат модули — соответствующие им подсети пятого уровня моделируют действия с таймерами. Общее число переходов сети, моделирующей спецификацию InRes, равно 266, а мест — 204.

В ходе экспериментов по симуляции этого протокола в системе SPV была подтверждена ошибка, описанная в работе [5].

5. ЗАКЛЮЧЕНИЕ

Автоматическая генерация сетевых моделей коммуникационных протоколов существенно сокращает трудоемкость проведения экспериментов по их верификации, а использование принципа иерархии — поуровневого создания сети — делает возможным построение сетевых моделей для систем реальной сложности. Моделирование протоколов посредством сетей Петри позволяет распознавать семантические ошибки, которые трудно обнаружить стандартными методами тестирования.

При помощи описанного транслятора были получены сетевые модели для кольцевого (Ring) и InRes протоколов, с полученными сетевыми моделями были проведены эксперименты. В ходе экспериментов были установлены семантические ошибки в этих протоколах.

СПИСОК ЛИТЕРАТУРЫ

1. **Карабегов А. В., Тер-Микаэлян Т. М.** Введение в язык SDL. — М.: Радио и связь, 1993. — 184 с.
2. **Specification and Description Language (SDL).** Recommendation Z.100. — ITU-T, 1993.
3. **Алексеев Г. И., Быстров А. В., Куртов С. А., Мыльников С. П., Непомнящий В. А., Окунишникова Е. В., Чубарев П. А., Чурина Т. Г.** Использование сетей Петри для верификации распределенных систем, представленных на языке

- Estelle // Известия РАН. Теория и системы управления. — 1999. — № 5. — С. 105–116.
4. **Churina T. G.** Colored Petri net approach to modeling of SDL specifications // Joint Bulletin of NCC & IIS. Ser.: Comput. Sci. — 2000. — N 13. — P. 18–39.
 5. **Fisher J., Dimitrov E.** Verification of SDL'92 specifications using extended Petri nets // Proc. IFIP 15th Intern. Symp. on Protocol Specification, Testing and Verification. — Warsaw, 1995. — P. 455–458.
 6. **Непомнящий В. А., Алексеев Г. И., Быстров А. В., Куртов С. А., Мыльников С. П., Окунишникова Е. В., Чубарев П. А., Чурина Т. Г.** Верификация Estelle-спецификаций распределенных систем посредством раскрашенных сетей Петри. — Новосибирск, 1998. — С. 103–112.