АВТОМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА, 1987, № 1, с. 9—15

УДК 681.324

Н. А. Анисимов

АЛГЕБРА СТРУКТУР ПРОТОКОЛОВ НА ОСНОВЕ ТЕОРИИ СЕТЕЙ ПЕТРИ

Введение. Успех создания программного обеспечения информационно-вычислительных сетей (ПО ИВС) [1] в значительной степени определяется наличием формальных методов, поддерживающих этапы его реализации. Распределенный характер, сложная иерархическая структура ПО ИВС, а также существующая тенденция к усложнению служб и протоколов обусловливают необходимость создания основ теории протоколов и в первую очередь разработку ее формального аппарата. Такой аппарат должен, во-первых, достаточно адекватно описывать структурные и поведенческие аспекты протоколов. Во-вторых, он должен обладать довольно развитой теорией, поддерживающей процессы формализации, верификации, синтеза и реализации протоколов.

В настоящее время накоплен довольно богатый опыт по формализации и верификации протоколов [2]. Большое внимание стало уделяться подходам, при которых сложные протоколы представляются как совокупность более простых протоколов. Известны методы композиции протоколов, использующие обобщенные автоматы [3], агрегаты [4], сети логических асинхронных процессов [5]. Для анализа корректности протоколов, наоборот, стали использоваться методы упрощения протоколов, среди которых необходимо выделить методы декомпозиции [6] и проекций протоколов [7].

В настоящей работе в качестве формальной основы теории протоколов предлагается использование теории сетей Петри [8], преимущества которых при описании асинхронных и параллельных процессов хорошо известны. Они обладают богатым арсеналом средств, к которым относятся методы анализа [8], многочисленные результаты моделирования вычислительных систем и сетевых протоколов [10].

Первоочередной задачей в процессе реализации ПО ИВС является задача формального описания всей совокупности протоколов. Характерной особенностью большинства современных протоколов и рекомендаций является разграничение в них аспектов управления с операциями над параметрами и переменными. Такое разделение выдвигает в качестве самостоятельной задачи задачу формализации структуры протоколов и многоуровневых систем протоколов.

Сложность реальных протоколов прежде всего обусловлена сложностью их структуры. К настоящему времени сложилась практика неформального описания протоколов по частям с разбивкой их на фазы, классы, подпротоколы и т. п. По-видимому, такой подход является перспективным и при формальном описании протоколов. Необходимы методы описания отдельных частей протоколов и формальные правила композиции их в единый протокол. Такой подход обеспечит наглядность и обозримость, сохраняя при этом строгость описания.

1. Алгебра структур протоколов. В настоящей работе принят подход [8], согласно которому все основные понятия и определения даются в терминах теории сетей Петри (ТСП), образуя таким образом протокольную интерпретацию ТСП. Это придаст теории протоколов формальный базис и позволит в дальнейшем переносить некоторые результаты ТСП в теорию протоколов.

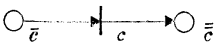


Рис. 1. Элементарная структура протокола.

Далее вводится алгебра протокольных структур, построенная на основе алгебры регулярных сетей [10], расширенной дополнительной операцией разрушения. Определения алгебры протокольных структур будут даваться в виде интерпретации элементов и операций алгебры регулярных сетей. Для краткости вместо термина «структура протокола», там, где это не вызовет недоразумений, будет употребляться термин «протокол».

Пусть зафиксировано конечное множество имен команд протокола $C=\{c_1, c_2, \ldots, c_n\}$. Каждое имя команды имеет префикс, которым может быть один из символов «↓» или «↑», обозначающий посылку или прием команды.

Элементарным протоколом будет называться сеть, вид которой представлен на рис. 1. Здесь $c \in C$, а $\bar{c}$ и $\bar{\bar{c}}$ — головное и хвостовое места сети, соответствующие начальному и конечному состоянию элементарного протокола. Операции над протоколами определяются как интерпретация аналогичных операций над регулярными сетями, которые хорошо известны [10] и поэтому будут представлены неформально. Каждому протоколу будет соответствовать регулярная сеть, головные и хвостовые места которой трактуются как начальное и конечное состояние протокола. Класс протоколов, порождаемый операциями над протоколами и классом элементарных протоколов, будет называться классом регулярных протоколов. Если при этом не используется операция наложения, то такой класс будет называться классом примитивных протоколов.

Пусть N, N_1 и N_2 — регулярные сети, а P, P_1 и P_2 — соответствующие им протоколы.

Операция итерации «*». Операция итерации над протоколом P_1 строит протокол $P=*(P_1)$ так, что $N=*(N_1)$. Протокол P цикличен, и его начальное состояние совпадает с конечным. Операция итерации позволяет описывать циклически выполняемые процедуры, характерные для всех протоколов.

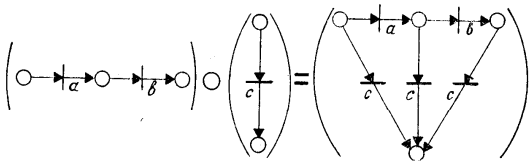
Операция присоединения «;». Операция присоединения протоколов P_1 и P_2 соединяет их в протокол $P=(P_1; P_2)$ так, что $N=(N_1; N_2)$. Начальным состоянием протокола P является начальное состояние протокола P_1, а конечным — конечное состояние протокола P_2. Операция присоединения формализует правило композиции двух протоколов, выполнение одного из которых может начаться только после завершения другого. Например, процедура передачи данных в протоколе может начаться только после завершения процедуры установления соединения.

Операция исключения «□». Операция исключения протоколов P_1 и P_2 объединяет их в протокол $P=(P_1 \square P_2)$ так, что $N=(N_1 \square N_2)$. При этом сливаются соответственно начальные и конечные состояния протоколов. Операция исключения строит конфигурацию с альтернативным выполнением процедур. С помощью этой операции описывается фундаментальная ситуация столкновения, имеющая место во многих протоколах.

Операция наложения «,». Операция наложения протоколов P_1 и P_2 объединяет их в протокол $P=(P_1, P_2)$ так, что $N=(N_1, N_2)$. При этом объединяются множества элементов сетей, соответствующих протоколам. Этой операцией можно описывать параллелизм, независимость выполнения процедур протоколов. Например, процедуры передачи срочных и нормальных блоков данных во многих протоколах независимы и могут быть представлены как параллельные процессы.

Для определения следующей операции необходимо обобщить определение операции наложения на случай, когда накладываемые сети

Рис. 2. Операция разрушения примитивной сети.



имеют кратные символы-переходы. Пусть в сети N_1 имеется k, а в сети N_2 — l переходов с символом a. Сначала в каждой сети нумеруются и помечаются индексом переходы с символами a: $\{a_1, a_2, \ldots, a_k\}$ и $\{a_1, a_2, \ldots, a_l\}$. Далее каждый переход a_i в сети N_1 копируется в l экземплярах и каждая копия помечается как a_i^j, где $j \in 1,l$. Аналогично копируются и помечаются a_i у сети N_2. После этого у всех переходов a_i^j сети N_2 верхний и нижний индексы меняются местами. После выполнения аналогичной последовательности действий над другими кратными символами-переходами, когда сети не имеют кратных переходов, выполняется операция наложения полученных сетей.

Операция разрушения «○». Операция разрушения не имеет аналога в алгебре регулярных сетей. Она вводится для описания распространенных процедур типа сброса, рестарта, разъединения и т. п., когда протокол может в любом состоянии выполнить требуемую процедуру. Аналогичная операция используется в работе [6] при декомпозиции протоколов.

Операция разрушения регулярных сетей определяется следующим образом. Пусть N_2 — сеть, имеющая единственное головное место, а сеть N_1 — примитивная, т. е. построенная без применения операции наложения. Сеть $N = (N_1 \bigcirc N_2)$ строится в два этапа. Сначала головное место сети N_2 и каждый инцидентный ему переход вместе с дугами копируются в количестве экземпляров, равном количеству мест в сети N_1. Затем каждое место сети N_1 сливается с одной из копий головного места сети N_2. Головным местом сети $N = (N_1 \bigcirc N_2)$ назначается головное место сети N_1, а хвостовым — хвостовое место сети N_2. Сеть N_1 будет называться разрушаемой, а сеть N_2 — разрушающей. На рис. 2 приведен пример операции разрушения.

Операция разрушения может быть определена для более широкого класса разрушаемых сетей, а именно, когда разрушаемая сеть представлена в виде $N_1 = (A, B)$, где регулярные сети A и B не имеют общих мест. Для этого сначала строятся сети $(A \bigcirc N_2)$ и $(B \bigcirc N_2)$, которые затем объединяются операцией наложения. На рис. 3 приведен пример разрушения сетей такого типа.

Операция разрушения протоколов P_1 и P_2 соединяет их в протокол $P = (P_1 \bigcirc P_2)$ так, что $N = (N_1 \bigcirc N_2)$. Начальным состоянием протокола P является начальное состояние протокола P_1, а конечным — конечное состояние протокола P_2.

Введенная алгебра, как и алгебра регулярных сетей Петри, допускает формульное представление, позволяя в компактной форме записывать сложные структуры протоколов. Полученные с помощью алгебры описания не только задают внешнее поведение протокольного объекта, но могут использоваться как определения его внутренней структуры, являясь одновременно и основой для реализации. Последнее, а также возможность описания процедур типа разрушения, отличает предлагаемую алгебру от других алгебраических подходов.

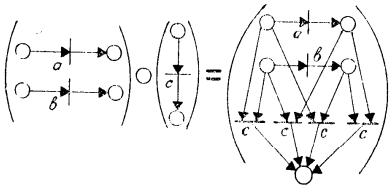


Рис. 3. Операция разрушения параллельной сети.

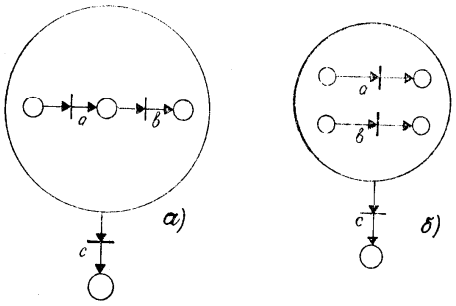


Рис. 4. Примеры макросетей.

2. Графическое представление структур протоколов. Несмотря на компактность формульного описания структуры, его графическое представление довольно громоздко. Сложность результирующей сети Петри обусловлена большим количеством переходов, порождаемых операцией разрушения. Для устранения этого недостатка предлагается структурированное представление сетей Петри этого класса, названное макросетями.

Все множество мест макросети разбивается на два непересекающихся множества простых и составных мест. Простые места аналогичны местам в сетях Петри. Составное место содержит внутри себя самостоятельную макросеть, допуская произвольную степень вложенности. Пусть задана сеть $N=(N_1 \bigcirc N_2)$. Графически она представляется в виде сети N_2, имеющей составное головное место, содержащее внутри себя сеть N_1. Результатом последовательного применения этого правила к каждой операции разрушения в формуле является макросеть, представляющая регулярную сеть с операцией разрушения.

Правила функционирования макросетей задаются следующим образом. Условием возбуждения перехода, имеющего входное составное место, является наличие не менее одной метки во внутренней сети. Если срабатывает переход, имеющий составное входное место, то из соответствующей внутренней сети извлекается метка, в каком бы она месте ни находилась. Если внутренняя сеть содержит параллельные фрагменты, то из каждого фрагмента извлекается по метке. Если переход имеет выходное составное место, то при его срабатывании метка помещается в головное место (места) внутренней сети.

На рис. 4—*а* и 4—*б* дано графическое представление сетей, представленных соответственно на рис. 2 и 3.

3. Пример. В качестве иллюстрации использования алгебры структур протоколов приводится структура пакетного уровня рекомендации X.25 для виртуальных каналов на стороне оконечного оборудования данных. Пусть $C=C_1 \cup C_2 \cup C_3 \cup C_4 \cup C_5 \cup C_6$, где $C_1=\{\downarrow CR, \uparrow CR, \downarrow CA, \uparrow CA\}$, $C_2=\{\downarrow CLR, \uparrow CLR, \downarrow CLC, \uparrow CLC\}$, $C_3=\{\downarrow DT, \uparrow DT, \downarrow RR, \uparrow RR, \downarrow RNR, \uparrow RNR\}$, $C_4=\{\downarrow I, \uparrow I, \downarrow IC, \uparrow IC\}$, $C_5=\{\downarrow RS, \uparrow RS, \downarrow RSA, \uparrow RSA\}$, $C_6=\{\downarrow RST, \uparrow RST, \downarrow RSTC, \uparrow RSTC\}$ — множества имен команд процедур установления соединения, разрыва соединения, управления потоком, передачи прерываний, сброса и рестарта соответственно. Структура процедур задается следующими формулами:

1. Установление соединения:

$$N_1=((\downarrow CR;\ (\uparrow CA \square (\uparrow CR;\ \uparrow CA)))\square(\uparrow CR;\ \downarrow CA)).$$

2. Разрыв соединения:

$$N_2=((\downarrow CLR;\ (\uparrow CLC \square \uparrow CLR))\square(\uparrow CLR;\ \downarrow CLC)).$$

3. Управление потоком:

$$N_3=(*(\downarrow DT \square \uparrow DT \square \downarrow RR \square \uparrow RR);\ *(\downarrow RNR;\ \downarrow RR);\ \ *(\uparrow RNR;\ \uparrow RR)).$$

4. Передача прерываний:

$$N_4=(*(\downarrow I;\ \uparrow IC),\ *(\uparrow I;\ \downarrow IC)).$$

5. Сброс:

$$N_5=((\downarrow RS;\ (\uparrow RS \square \uparrow RSA))\square(\uparrow RS;\ \downarrow RSA)).$$

6. Рестарт:

$$N_6=((\downarrow RST;\ (\uparrow RSTC \square \uparrow RST))\square(\uparrow RST;\ \downarrow RSTC)).$$

Структура всей процедуры выражается формулой

$$N = \bullet(\bullet((N_1;\bullet((N_3, N_4)\bigcirc N_5))\bigcirc N_2)\bigcirc N_6).$$

На рис. 5 дано графическое представление рассмотренной структуры. В работе [11] также представлено описание структуры транспортного протокола.

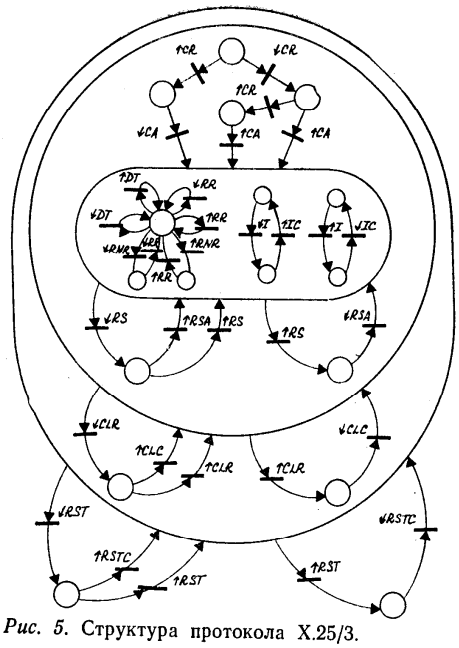


Рис. 5. Структура протокола X.25/3.

4. Корректность структур протоколов. Свойства протоколов во многом определяются свойствами их структур. Поэтому актуальна задача анализа корректности протокольных структур. Алгебраический подход позволяет свести анализ корректности сложных структур к анализу их составляющих.

Для анализа протокола на корректность необходимо построить модель, состоящую из протокольных объектов и среды. Структура протокола, определенная с помощью регулярной сети, задает поведение протокольного объекта. Дальнейшее изложение будет опираться на модель и результаты теории проекций протоколов [7]. Исходя из терминологии этой теории операция разрушения является хорошо сформированной, поскольку согласно определению состояние разрушения протокольного объекта непосредственно достижимо из любых состояний разрушаемого протокола. Как следует из основного результата теории проекций, хорошая сформированность гарантирует совпадение свойств живости и безопасности у всего протокола и у его составляющих. Таким образом, задача анализа этих свойств у протокола, заданного операцией разрушения, сводится к анализу свойств разрушаемого и разрушающего протоколов.

Для регулярного протокола определяется понятие проекции относительно степени вложенности. Проекцией первой степени протокола P называется исходная сеть, у которой все составные места заменены на простые. Проекцией степени k, где $k>1$, называется сеть, являющаяся внутренней сетью для проекции степени, у которой собственные составные места заменены на простые. В силу хорошей сформированности протокола его анализ на наличие свойств живости и безопасности сводится к анализу этих свойств у всех его проекций.

Таким образом, чем выше степень структурированности, тем на большее число более простых составляющих разбивается протокол и, следовательно, тем проще их анализ.

5. Иерархическая композиция протоколов. ПО ИВС имеет иерархическую структуру и разбивается на уровни, каждый из которых функционирует в соответствии с собственным протоколом. Совокупность организованных таким образом протоколов образует иерархическую систему протоколов. С другой стороны, в рамках одного уровня бывает целесообразно деление на подуровни и соответственно разбиение протокола на подпротоколы. Это обусловливает необходимость формализации понятия иерархической зависимости протоколов и введение формальных правил иерархической композиции структур протоколов.

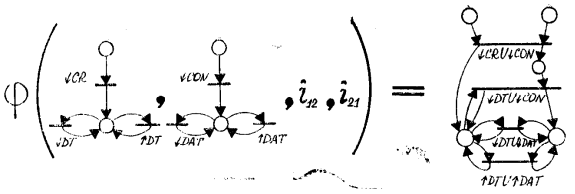


Рис. 6. Операция иерархической композиции.

Неформально иерархическая зависимость протоколов проявляется в том, что выполнению команды протокола одного уровня соответствует выполнение одной или последовательности команд другого уровня. Пусть даны два протокола P_1 и P_2, определенные на множестве имен команд C_1 и C_2. Иерархическая зависимость между протоколами P_1 и P_2 задается отображениями $i_{12}: C_1 \to F(C_2)$ и $i_{21}: C_2 \to F(C_1)$, где $F(C)$ обозначает все множество примитивных формул алгебры протоколов на множестве имен C. Действительно, команде одного протокола сопоставляется определенная структура, построенная из команд другого протокола, которая задает порядок на выполнение.

Операция иерархической композиции протоколов P_1 и P_2 относительно отображений i_{12} и i_{21} объединяет их в протокол $P=\varphi(P_1, P_2, i_{12}, i_{21})$, который строится по следующим правилам. Сначала в формуле протокола P_1 каждое вхождение символа $c_1 \in C_1$ заменяется на формулу $i_{12}(c_1)$, в которой вместо каждого имени $c_2 \in C_2$ подставляется имя $c_1 \cup c_2$. Здесь сделано простое обобщение, позволяющее наравне с именами команд оперировать и с их множествами. Аналогично преобразуется и формула протокола P_2. И, наконец, сети, соответствующие полученным формулам, объединяются операцией наложения.

На рис. 6 представлен пример выполнения операции иерархической композиции простых протоколов $P_1=(\downarrow CR;\ *(\downarrow DT \square \uparrow DT))$ и $P_2=(\downarrow CON;\ *(\downarrow DAT \square \uparrow DAT))$ относительно отображений i_{12} и i_{21}:

$$i_{12}:\ \downarrow CR \to \downarrow CON \qquad\qquad i_{21}:\ \downarrow CON \to (\downarrow CR;\ \downarrow DT)$$

$$\downarrow DT \to (\downarrow CON \square \downarrow DAT) \qquad\qquad \downarrow DAT \to \downarrow DT$$

$$\uparrow DT \to \uparrow DAT \qquad\qquad \uparrow DAT \to \uparrow DT$$

Передача и прием всех команд протокола P_2 производятся посредством команд $\downarrow DT$ и $\uparrow DT$ протокола P_1. Кроме того, выполнению команды $\downarrow CON$ протокола P_2 должно соответствовать последовательное выполнение команд $\downarrow CR$ и $\downarrow DT$ протокола P_1.

Операция иерархической композиции протоколов приводит к тому, что срабатыванию одного перехода может соответствовать выполнение сразу двух команд различного уровня. Это избавляет от необходимости разработки и реализации специального интерфейса и позволяет многоуровневую логическую структуру ПО ИВС описывать одной сетью и реализовать в едином программном (аппаратном) модуле.

Заключение. В работе предложена алгебра, позволяющая получать строгое, компактное и обозримое описание структур сложных протоколов. Преимуществом алгебраического подхода является то, что происходит сближение процессов разработки, формализации и осмысления протоколов, т. е. процесс осмысления формального описания протокола воспроизводит процесс его разработки и формализации.

В плане верификации протоколов предложенный метод позволяет сводить задачу верификации сложных протоколов к верификации его составляющих. Схожесть структур различных протоколов, по-видимому, позволит выделить набор стандартных «правильных» структур и конструировать из них протоколы с заведомо хорошими свойствами.

СПИСОК ЛИТЕРАТУРЫ

1. *Якубайтис Э. А.* Информационно-вычислительные сети. М.: Финансы и статистика, 1984. 232 с.
2. *Зайцев С. С., Мурадян Н. А.* Методы проверки правильности функционирования сетевых протоколов. М., 1983. 68 с. (Препринт/Научн. совет по комплексной проблеме «Кибернетика» АН СССР).
3. *Лазарев В. Г.* Элементы автоматной теории протоколов сетей ЭВМ. — В кн.: Труды IV Всесоюзной школы-семинара по проблемам управления на сетях и узлах связи. М.: Наука, 1984, с. 68—71.
4. *Пранявичус Г. И.* Модели и методы исследования вычислительных систем. Вильнюс: Мокслас, 1982. 288 с.
5. *Ротанов С. В.* Об одном подходе к анализу корректности протоколов. — АВТ, 1982, № 4, с. 17—19.
6. *Vuong S. T., Cowan D. D.* A decomposition method for the validation of structured protocols. — INFOCOM 82. Proc. Conf., p. 209—220.
7. *Lam S. S., Shankar A. U.* Protocol verification via projections. — IEEE Trans. Software Eng., 1984, vol. SE-10, N 4, p. 325—342.
8. *Питерсон Дж.* Теория сетей Петри и моделирование систем. М.: Мир, 1984. 264 с.
9. *Анисимов Н. А.* Методы формального описания, верификации и реализации сетевых протоколов с использованием теории сетей Петри. Владивосток, 1984. 40 с. (Препринт/ИАПУ ДВНЦ АН СССР, № 118).
10. *Котов В. Е.* Алгебра регулярных сетей Петри. — Кибернетика, 1980, № 5, с. 10—18.
11. *Анисимов Н. А.* Алгебраический подход к формализации и верификации структуры протоколов на основе теории сетей Петри. — В кн.: Вычислительные сети коммутации пакетов: Тез. докл. IV Всесоюз. конф. Ч. 1. Рига, 1985, с. 158—162.

Поступила в редакцию
10.12.85