Методы описания и анализа распределенных систем

Пособие подготовлено Жуковым Д.Ю.

E-mail: dmjr@ispras.ru

Версия от 23 декабря 2000

Содержание

1 Введение **1**
- 1.1 Использованные источники 1

2 Графовые модели распределенных систем **1**
- 2.1 Размеченные системы переходов и процессные графы 1
- 2.2 Операции над процессными графами 2
- 2.3 Сильная эквивалентность на размеченных системах переходов 5
- 2.4 Свойства сильной эквивалентности 6
- 2.5 Распознавание сильной эквивалентности модальными формулами 8
- 2.6 Алгоритм построения сильной эквивалентности 10
- 2.7 Минимизация процессных графов 12

3 Исчисление взаимодействующих систем (CCS) **15**
- 3.1 Синтаксис CCS 15
- 3.2 Сорта процессных выражений 18
- 3.3 Эквивалентные преобразования процессов 19
- 3.4 Сохранение сильной эквивалентности операциями построения процессов 21
- 3.5 Наблюдаемая эквивалентность 25

Список литературы **26**

1 Введение

Настоящее учебное пособие посвящено математическим методам и анализам распределенных систем. Методы, изложенные в нем, пригодны для исследования как программных, так и аппаратных систем.

Данная версия текста является предварительной и может содержать ошибки. О них можно сообщать по адресу `dmjr@ispras.ru`. Замечания и предложения читателей могут быть очень полезны автору в его работе над текстом.

1.1 Использованные источники

Многие результаты, приведенные в этом пособии, заимствованы из книги [Mil89]. Автор пособия иногда использовал терминологию и обозначения, отличные от использованных в этом источнике. Так, вместо термина "*агент CCS*" автор использовал термин "*процесс*". Некоторые утверждения, приведенные в [Mil89] для термов CCS, сформулированы автором для произвольных размеченных систем переходов. При введении определений, формулировании и доказательстве утверждений в пособии приведены ссылки на аналогичные утверждения из других работ. Ссылки даны в скобках или в сносках.

2 Графовые модели распределенных систем

2.1 Размеченные системы переходов и процессные графы

Определение 1 Пусть задано некоторое множество Act, которое мы будем называть *множеством действий. Размеченной системой переходов*[1] (РСП) над Act называется пара (S, R), где

- S – *множество состояний.*
- R – *множество переходов.* Каждый переход представляет из себя тройку (s_1, α, s_2), где $s_1, s_2 \in S$ и $\alpha \in Act$. Если в множестве R содержится такая тройка, мы будем говорить, что из состояния s_1 возможен переход в состояние s_2 с выполнением действия α. Если размеченная система переходов L фиксированна, то факт $(s_1, \alpha, s_2) \in R$ мы будем обозначать записью $s_1 \xrightarrow{\alpha} s_2$. □

Определение 2 Пусть задано множество действий Act. *Процессным графом* над Act называется тройка $G = (S, s_0, R)$, где

- (S, R) – размеченная система переходов над Act.
- $s_0 \in S$ – *начальное состояние* графа. □

В дальнейшем мы будем предполагать, что множество Act фиксировано, и рассматривать только РСП и процессные графы над этим множеством действий.

[1] В англоязычной литературе такая структура называется labeled transition system (LTS). В настоящем пособии будет использоваться русская аббревиатура РСП.

Определение 3 Состояние $s \in S$ мы будем называть *достижимым* в графе $G = (S, s_0, R)$, если в этом графе существует путь из состояния s_0 в состояние s. □

Недостижимые состояния не дают никакой информации о поведении описываемой системы.

Соглашение: Мы будем считать графы G_1 и G_2 эквивалентными, если они удовлетворяют одному из следующих условий:

1. Графы G_2 и G_1 изоморфны (то есть G_2 можно получить из G_1 переименованием состояний).

2. Граф G_2 можно получить из G_1 добавлением и удалением недостижимых состояний. □

В разделе 2.3 мы рассмотрим определение эквивалентности графов, которое позволит отождествлять графы при выполнении условий 1 и 2. При этом эквивалентными будут не только пары графов, которые описываются этими двумя условиями.

Определение 4 Пусть дан процессный граф $G = (S, s_0, R)$. ***Вычислением*** (или ***протоколом***) процессного графа G называется слово $\alpha_1 \dots \alpha_n \in Act^*$ в алфавите Act, для которого существует такой путь $s_0, \dots, s_n$ в графе G, что $s_i \xrightarrow{\alpha_{i+1}} s_{i+1}$ для любого i. Множество всех вычислений графа G мы будем обозначать через $Comp(G)$. □

Словом NIL мы будем обозначать тривиальный граф, то есть граф, не содержащий никаких переходов. Множество его состояний состоит из единственного состояния, которое является начальным.

$$NIL \stackrel{def}{=} (\{s_0\}, s_0, \emptyset)$$

2.2 Операции над процессными графами

Теперь мы введем ряд операций над процессными графами, при помощи которых из одних графов можно строить другие.

Определение 5 Пусть задан процессный граф $G = (S, s_0, R)$ и некоторое действие $\alpha \in Act$. Результат *добавления префикса* α к графу G – это граф $\alpha.G$, определяемый следующим образом:

$$\alpha.G \stackrel{def}{=} \big(S \cup \{s_0'\}, s_0', R \cup \{(s_0', \alpha, s_0)\}\big),$$

где s_0' – новая вершина, не принадлежащая множеству S. □

Иными словами, граф $\alpha.G$ получается из графа G добавлением

- новой вершины s_0' и

- нового перехода из вершины s_0' в вершину s_0 с выполнением действия α.

Определение 6 Пусть заданы два процессных графа $G_1 = (S^{(1)}, s_0^{(1)}, R^{(1)})$ и $G_2 = (S^{(2)}, s_0^{(2)}, R^{(2)})$. Пусть также множества $S^{(1)}$ и $S^{(2)}$ не пересекаются, то есть $S^{(1)} \cap S^{(2)} = \emptyset$. Тогда *альтернативной композицией* графов G_1 и G_2 мы будем называть граф $G_1 + G_2 \stackrel{def}{=} (S, s_0, R)$ со следующими компонентами:

- $S = S^{(1)} \cup S^{(2)} \cup \{s_0\}$, где $s_0 \notin S^{(1)} \cup S^{(2)}$.
- $R = R^{(1)} \cup R^{(2)} \cup \{(s_0, \alpha, s) \mid (s_0^{(1)}, \alpha, s) \in R^{(1)}\} \cup$
  $\cup \{(s_0, \alpha, s) \mid (s_0^{(2)}, \alpha, s) \in R^{(2)}\}$.

Если графы G_1 и G_2 имеют общие состояния, то их альтернативная композиция по определению равна $G_1 + G_2'$, где G_2' – граф, полученный из G_2 переименованием состояний и не имеющий общих состояний с G_1. □

Отметим, что если графы G_1 и G_2 в определении 6 не содержат недостижимых состояний, то в некоторых случаях может получиться, что состояние $s_0^{(1)}$ или $s_0^{(2)}$ будет недостижимо в графе $G_1 + G_2$.

В дальнейших рассуждениях мы будем предполагать, что множество Act устроено особым образом. Введем множество *имен действий* $\mathcal{N}$. Множество *коимен* $\overline{\mathcal{N}}$ состоит из элементов $\bar{a}$ для всех $a \in \mathcal{N}$:

$$\overline{\mathcal{N}} \stackrel{def}{=} \{\bar{a} \mid a \in \mathcal{N}\}$$

Множество *меток* $\mathcal{L}$ состоит из всех имен и коимен:

$$\mathcal{L} \stackrel{def}{=} \mathcal{N} \cup \overline{\mathcal{N}}$$

Множество действий Act определяется как множество $\mathcal{L}$, к которому добавлено особое действие τ (при этом $\tau \notin \mathcal{N}$):

$$Act \stackrel{def}{=} \mathcal{L} \cup \{\tau\}$$

Действие τ называется *внутренним* или *невидимым* действием. Оно обозначает действие некоторого объекта, не связанное со взаимодействием с другими объектами. Обычно мы будем использовать имена для обозначения действий по *приему* информации или получению предметов. Коимена обычно будут обозначать действия по *выдаче* информации или предметов.

На множестве $\mathcal{L}$ мы зададим функцию $\bar{\cdot} : \mathcal{L} \to \mathcal{L}$. Для каждого $a \in Names$ ее значение на a есть $\bar{a}$, а значение на $\bar{a}$ есть a. Таким образом, для любого $\alpha \in \mathcal{L}$ выполнено соотношение $\bar{\bar{\alpha}} = \alpha$.

Соглашение: Имена действий (то есть элементы множества $\mathcal{N}$) мы будем обозначать латинскими буквами $(a, b, c, \ldots)$. Действия (то есть элементы Act) мы будем обозначать греческими буквами $(\alpha, \beta, \gamma, \ldots)$. □

Определение 7 Пусть заданы два процессных графа $G_1 = (S^{(1)}, s_0^{(1)}, R^{(1)})$ и $G_2 = (S^{(2)}, s_0^{(2)}, R^{(2)})$. *Параллельной композицией* графов G_1 и G_2 называется граф $G_1 | G_2 \stackrel{def}{=} (S, s_0, R)$, имеющий следующие компоненты:

- $S = S^{(1)} \times S^{(2)}$.

- $s_0 = (s_0^{(1)}, s_0^{(2)})$.
- Множество R есть минимальное множество, удовлетворяющее следующим условиям:

  1. $(s_1, \alpha, s_1') \in R^{(1)}, s_2 \in S^{(2)} \Rightarrow \big((s_1, s_2), \alpha, (s_1', s_2)\big) \in R$;
  2. $(s_2, \alpha, s_2') \in R^{(2)}, s_1 \in S^{(1)} \Rightarrow \big((s_1, s_2), \alpha, (s_1, s_2')\big) \in R$;
  3. $(s_1, \alpha, s_1') \in R^{(1)}, (s_2, \bar{\alpha}, s_2') \in R^{(2)}, \alpha \neq \tau \Rightarrow$
     $\Rightarrow \big((s_1, s_2), \tau, (s_1', s_2')\big) \in R.$ □

Определение 8 Пусть задан процессный граф $G = (S, s_0, R)$ и некоторое множество имен действий $L \subseteq \mathcal{N}$. *Ограничением* графа G по множеству L называется граф $G \setminus L \stackrel{def}{=} (S, s_0, R')$, где

$$R' = \{(s_1, \alpha, s_2) \mid (s_1, \alpha, s_2) \in R,\ \alpha, \bar{\alpha} \notin L\}$$

□

Определение 9 *Функцией переименования* мы будем называть отображение $f : \mathcal{L} \to \mathcal{L}$, сохраняющее функцию $\bar{\cdot}$. Последнее значит, что для любого действия $\alpha \in \mathcal{L}$ должно быть выполнено соотношение $f(\bar{\alpha}) = \overline{f(\alpha)}$. □

Для каждой функции переименования f мы определим ее продолжение, отображающее множество Act в себя. На действии τ мы определим продолжение f следующим образом: $f(\tau) = \tau$. Это продолжение функции f мы будем обозначать тем же символом f.

Определение 10 Пусть задан процессный граф $G = (S, s_0, R)$ и функция переименования f. Результатом *переименования* действий графа G согласно функции f мы будем называть граф $G[f] \stackrel{def}{=} (S, s_0, R')$, где

$$R' = \{(s_1, f(\alpha), s_2) \mid (s_1, \alpha, s_2) \in R\}$$

□

Пусть имена действий $a_1, \ldots, a_n$, действия $\beta_1, \ldots, \beta_n$ и функция переименования f таковы, что

- действия $a_1, \ldots, a_n$ попарно различны;
- $\forall i \, : \, f(a_i) = \beta_i$;
- $\forall a \in \mathcal{N} \setminus \{a_1, \ldots, a_n\} \, : \, f(a) = a$.

В этом случае мы будем часто использовать для графа $G[f]$ альтернативную запись $G[\beta_1/a_1, \ldots, \beta_n/a_n]$.

2.3 Сильная эквивалентность на размеченных системах переходов

Поставим вопрос о том, в каких случаях имеет смысл считать, что два графа определяют одно и то же поведение. Такая задача может появиться, например, если мы хотим хранить процессные графы в памяти – тогда из всевозможных графов нам хотелось бы уметь выбирать граф с минимальным числом состояний, определяющий, тем не менее, некоторое заданное поведение.

Определим вначале эквивалентность состояний в размеченных системах переходов:

Определение 11 Пусть дана размеченная система переходов $L = (S, R)$. Бинарное отношение $E \subseteq R$ называется *сильным бимоделированием* на L, если для любых двух состояний $(s_1, s_2) \in S$ выполнены следующие условия:

1. Если действие α и состояние s_1', таковы, что $s_1 \xrightarrow{\alpha} s_1'$, то существует такое состояние s_2', что $s_2 \xrightarrow{\alpha} s_2'$ и $(s_1', s_2') \in E$.
2. Второе условие симметрично первому:

$$\forall \alpha \in Act,\ s_2' \in S\ :\ s_2 \xrightarrow{\alpha} s_2' \ \Rightarrow\ \exists s_1' \in S\ :\ s_1 \xrightarrow{\alpha} s_1' \ \wedge\ (s_1', s_2') \in E$$

Если $s_1, s_2 \in S$ – два состояния, то они называются *сильно эквивалентными*, если существует хотя бы одно сильное бимоделирование, содержащее пару (s_1, s_2). Сильная эквивалентность состояний s_1 и s_2 обозначается через $s_1 \sim s_2$.

Отношение сильной эквивалентности состояний РСП L мы иногда будем обозначать через $\sim_L$. Аналогично, $\sim_G$ иногда будет обозначать отношение сильной эквивалентности состояний процессного графа G. □

В дальнейшем мы часто будем использовать термин "сильное бимоделирование" без указания РСП, если из контекста будет понятно, о какой системе идет речь. Эквивалентность процессных графов определяется следующим образом:

Определение 12 Пусть заданы два процессных графа $G_1 = (S^{(1)}, s_0^{(1)}, R^{(1)})$ и $G_2 = (S^{(2)}, s_0^2, R^{(2)})$. Пусть множества их состояний не пересекаются: $S^{(1)} \cap S^{(2)} = \emptyset$. В этом случае графы G_1 и G_2 *сильно эквивалентны*, если $s_0^{(1)} \sim s_0^{(2)}$ в РСП $(S^{(1)} \cup S^{(2)}, R^{(1)} \cup R^{(2)})$. Сильная эквивалентность графов G_1 и G_2 обозначается через $G_1 \sim G_2$.

Если множества состояний графов G_1 и G_2 пересекаются, то по определению $G_1 \sim G_2 \Leftrightarrow G_1 \sim G_2'$, где G_2' – граф, полученный из G_2 переименованием состояний и не имеющий общих состояний с G_1. □

Упражнение 1 Доказать, пользуясь определением сильной эквивалентности 12, что графы G_1 и G_2 на рисунке 1 не сильно эквивалентны, в то время как графы $G_1' = G_1[b/c]$ и $G_2' = G_2[b/c]$ на рисунке 2 сильно эквивалентны. Обратите внимание, что множества вычислений графов G_1 и G_2 равны, хотя эти графы не сильно эквивалентны[2]:

[2]Символом ε здесь обозначена пустая строка.

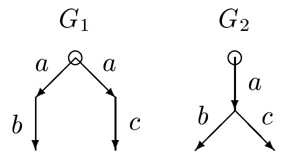

Рис. 1: Два процессных графа

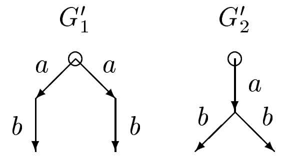

Рис. 2: Процессные графы с переименованными действиями

$$Comp(G_1) \quad = \quad \{\varepsilon, a, ab, ac\} \qquad\qquad Comp(G_2) \quad = \quad \{\varepsilon, a, ab, ac\}$$

$\square$

2.4 Свойства сильной эквивалентности

Рассмотрим ряд свойств отношения сильной эквивалентности. Для доказательства этих свойств нам потребуется следующее утверждение[3]:

Утверждение 1 *Пусть $L = (S, R)$ есть размеченная система переходов. Тогда выполнены следующие утверждения:*

1. *Тождественное отношение Id_S на множестве S является сильным бимоделированием.*

2. *Если $E \subseteq S^2$ – сильное бимоделирование, то E^{-1} – также сильное бимоделирование.*

3. *Если отношения $E_1, E_2 \subseteq S^2$ – сильные бимоделирования, то их произведение $E_1 \circ E_2$ – также сильное бимоделирование.*

4. *Для произвольного множества I и любого семейства сильных бимоделирований $\{E_i \mid i \in I\}$ сильным бимоделированием будет отношение $\bigcup_{i \in I} E_i$.*

В этом утверждении словосочетание "сильное бимоделирование" везде обозначает сильное бимоделирование на L.

Доказательство. *Пункт 1.* Если пара $(s_1, s_2) \in Id_S$, то $s_1 = s_2$. В первом и втором условиях из определения 11 необходимо взять $s_1' = s_2'$.

Пункт 2. Если $(s_1, s_2) \in E^{-1}$, то $(s_2, s_1) \in E$. Первое условие из определения 11 для пары (s_1, s_2) и отношения E^{-1} выполнено, так как выполнено

[3] Аналогичное утверждение приведено в работе [Mil89] на странице 90 для процессов CCS. Исчисление CCS будет введено в разделе 3.

второе условие для пары (s_2, s_1) и отношения E. Аналогично проверяется и второе условие для пары (s_1, s_2) и отношения E^{-1}.

Пункт 3. Пусть $(s_1, s_2) \in E_1 \circ E_2$. Докажем, что первое условие из определения 11 выполнено для этой пары. Пусть $\alpha \in Act$, $s_1' \in S$ и $s_1 \xrightarrow{\alpha} s_1'$. Укажем такое $s_2' \in S$, что $s_2 \xrightarrow{\alpha} s_2'$ и $(s_1', s_2') \in E_1 \circ E_2$.

По определению отношения $E_1 \circ E_2$ существует такое $s_3 \in S$, что $(s_1, s_3) \in E_1$ и $(s_3, s_2) \in E_2$. Так как E_1 – сильное бимоделирование, существует такое $s_3' \in S$, что $s_3 \xrightarrow{\alpha} s_3'$ и $(s_1', s_3') \in E_1$. Далее, так как E_2 – сильное бимоделирование, существует такое $s_2' \in S$, что $s_2 \xrightarrow{\alpha} s_2'$ и $(s_3', s_2') \in E_1$. Полученное состояние s_2' таково, что $(s_1', s_2') \in E_1 \circ E_2$. Таким образом, подходящее состояние s_2' найдено. Второе условие из определения 11 для пары (s_1, s_2) и отношения $E_1 \circ E_2$ проверяется аналогично.

Пункт 4. Пусть $E = \bigcup_{i \in I} E_i$, $(s_1, s_2) \in E$. Проверим первое условие из определения 11 для пары (s_1, s_2). Пусть $\alpha \in Act$, $s_1' \in S$ и $s_1 \xrightarrow{\alpha} s_1'$. Имеем $(s_1, s_2) \in E_i$ для некоторого $i \in I$. Отношение E_i является сильным бимоделированием, а значит существует такое $s_2' \in S$, что $s_2 \xrightarrow{\alpha} s_2'$ и $(s_1', s_2') \in E_i$. При этом $(s_1', s_2') \in E$, что доказывает выполнение первого условия. Второе условие проверяется аналогично. □

Пользуясь этим утверждением, можно доказать ряд свойств сильной эквивалентности[4].

Утверждение 2 *Пусть $L = (S, R)$ – размеченная система переходов. Тогда отношение $\sim$ сильной эквивалентности состояний L обладает следующими свойствами:*

1. *$\sim$ является отношением эквивалентности на S.*
2. *$\sim$ является сильным бимоделированием на L.*
3. *$\sim$ является максимальным сильным бимоделированием на L, то есть $E \subseteq L$ для любого сильного бимоделирования E на L.*

Доказательство. Докажем вначале пункт 2. Заметим, что отношение $\sim$ представимо в следующем виде:

$$\sim = \bigcup \left\{ E \subseteq S^2 \mid E \text{ – сильное бимоделирование на } L \right\} \tag{1}$$

По пункту 4 утверждения 1 отсюда следует, что $\sim$ – сильное бимоделирование.

Пункт 3 утверждения 2 следует из (1) очевидным образом.

Докажем теперь пункт 1.

Рефлексивность: Пусть $s \in S$. Отношение Id_S является сильным бимоделированием и содержит пару (s, s), следовательно $s \sim s$.

Симметричность: Пусть $s_1 \sim s_2$. В этом случае $(s_2, s_1) \in \sim^{-1}$. По пункту 2 утверждения 1 отношение $\sim^{-1}$ является сильным бимоделированием, а значит $s_2 \sim s_1$.

Транзитивность: Пусть $s_1 \sim s_2$ и $s_2 \sim s_3$. Тогда $(s_1, s_3) \in \sim \circ \sim$. По пункту 3 утверждения 1 отношение $\sim \circ \sim$ является сильным бимоделированием, а значит $s_1 \sim s_3$. □

[4]Эти свойства приведены приведены и доказаны в работе [Mil89] на страницах 90 – 91 для процессов CCS.

2.5 Распознавание сильной эквивалентности модальными формулами

Для того, чтобы доказать сильную эквивалентность двух состояний s_1 и s_2, достаточно указать хотя бы одно сильное бимоделирование на системе переходов, содержащее пару (s_1, s_2). Теорема, которая будет доказана в этом разделе, дает критерий сильной эквивалентности состояний. В некоторых случаях она позволяет удобно доказывать *неэквивалентность* двух данных состояний в некоторой системе переходов.

Данная теорема основана на использовании логики Хеннеси-Милнера. Эта логика относится к модальным логикам. Она позволяет описывать *локальные* свойства состояний систем переходов. В ней имеются специальные конструкции, позволяющие для данного состояния задавать условия на состояния, связанные с ним. Логика Хеннеси-Милнера содержит и обычные булевы связки: конъюнкцию, дизъюнкцию, отрицание, и т. д., – и константы истина и ложь.

Определение 13 Пусть задано множество действий Act. Множество *формул* логики Хеннеси-Милнера мы обозначим символом Fm. Это множество по определению есть минимальное множество, обладающее следующими свойствами:

1. Константа $\top$ является формулой: $\top \in Fm$.

2. Для любых двух формул $f_1,\ f_2 \in Fm$, выражения $f_1 \wedge f_2$ и $\neg f_1$ тоже будут формулами.

3. Для любой формулы $f \in Fm$ и действия $\alpha \in Act$ выражение $\langle\alpha\rangle f$ также будет формулой. Формула $\langle\alpha\rangle f$ читается как “ромб α f”. □

Другие привычные связки, такие как дизьюнкция и импликация, и константу ложь мы будем рассматривать как сокращения. Также мы введем сокращение $[\alpha]$ как связку, двойственную к $\langle\alpha\rangle$:

$$
\begin{array}{rclcrcl}
\bot & \stackrel{def}{=} & \neg\top & \qquad & f_1 \vee f_2 & \stackrel{def}{=} & \neg(\neg f_1 \wedge \neg f_2) \\
f_1 \to f_2 & \stackrel{def}{=} & \neg f_1 \vee f_2 & \qquad & [\alpha] f & \stackrel{def}{=} & \neg\langle\alpha\rangle\neg f
\end{array}
$$

Формула $[\alpha] f$ читается как “бокс α f”. Каждая формула описывает некоторое свойство состояний в системе переходов. В каждом состоянии отдельная формула может быть либо истинной, либо ложной. Если формула f истинна в состоянии s, мы будем писать $s \models f$.

Определение 14 Пусть $L = (S, R)$ есть размеченная система переходов. Истинность формулы $f \in Fm$ в состоянии $s \in S$ определяется индукцией по структуре f:

- Формула $\top$ истинна в любом состоянии $s \in S$: $\forall s \in S\ :\ s \models \top$.

- Формула $\neg f$ истинна в состоянии $s \in S$ тогда и только тогда, когда в состоянии s формула f ложна: $s \models \neg f \stackrel{def}{\Longleftrightarrow} s \not\models f$.

- $s \models f_1 \wedge f_2 \stackrel{def}{\Longleftrightarrow} s \models f_1 \wedge s \models f_2$.
- $s \models \langle\alpha\rangle f \stackrel{def}{\Longleftrightarrow} \exists s' \in S : s \stackrel{\alpha}{\longrightarrow} s' \wedge s' \models f$. □

Пример 1 Пусть $\alpha \in Act$ и s – состояние в некоторой РСП. Истинность формулы $\langle a\rangle \top$ в состоянии s означает, что существует хотя бы один переход из состояния s с выполнением действия a. □

Определение 15 Размеченная система переходов (S, R) является *конечно-ветвящейся*, если для любых действия $\alpha \in Act$ и состояния $s \in S$ множество

$$\{s' \mid s \stackrel{\alpha}{\longrightarrow} s'\}$$

конечно. □

Применять формулы модальной логики для распознавания сильной эквивалентности графов позволяет следующая теорема[5]:

Теорема 3 *Пусть $L = (S, R)$ есть размеченная система переходов, а $s_1, s_2 \in S$ – два состояния. Тогда*

1. *Если $s_1 \sim s_2$, то в s_1 и s_2 истинны одни и те же формулы логики Хеннеси-Милнера:*

$$s_1 \sim s_2 \Rightarrow \forall f \in Fm : (s_1 \models f \Leftrightarrow s_2 \models f)$$

2. *Если L конечно-ветвящаяся и $\forall f \in Fm : (s_1 \models f \Leftrightarrow s_2 \models f)$, то $s_1 \sim s_2$.*

Доказательство. *Пункт 1.* Пусть $f \in Fm$. Индукцией по структуре f покажем, что если $s_1 \sim s_2$, то $s_1 \models f \Leftrightarrow s_2 \models f$.

- Если $f = \top$, то доказываемое утверждение очевидно.
- Пусть $f = \neg g$. Формула g имеет меньшую сложность, нежели f, поэтому для нее верно предположение индукции. Получаем

$$s_1 \models f \Leftrightarrow s_1 \not\models g \Leftrightarrow s_2 \not\models g \Leftrightarrow s_2 \models f$$

- В случае $f = g_1 \wedge g_2$ рассуждения аналогичны доказательству в предыдущем пункте.
- Пусть $f = \langle\alpha\rangle g$ и $s_1 \models f$. Докажем, что $s_2 \models f$. Так как $s_1 \models \langle\alpha\rangle g$, то существует $s_1' \in S$, такое что $s_1 \stackrel{\alpha}{\longrightarrow} s_1'$ и $s_1' \models g$. Поскольку $\sim$ является сильным бимоделированием, из $s_1 \sim s_2$ следует существование такого $s_2' \in S$, что $s_2 \stackrel{\alpha}{\longrightarrow} s_2'$ и $s_1' \sim s_2'$. Применяя предположение индукции к формуле g, мы получим $s_2' \models g$. Отсюда следует $s_2 \models \langle\alpha\rangle g$, то есть $s_2 \models f$.

  Мы доказали, что из $s_1 \models f$ следует $s_2 \models f$. Обратное следствие доказывается аналогично.

[5] Рассуждения, использованные при доказательстве этой теоремы, в большой мере заимствованы из книги [Mil89]. В этой книге на страницах 228 – 229 доказана теорема, аналогичная данной, для термов CCS, а не для произвольных РСП.

Итак, пункт 1 доказан.

Пункт 2. Пусть РСП L конечно-ветвящаяся, и в состояниях s_1 и s_2 истинны одни и те же формулы логики Хеннеси-Милнера. Определим отношение E на S следующим образом:

$$E = \{(s_1, s_2) \mid \forall f \in Fm \, : \, s_1 \models f \Leftrightarrow s_2 \models f\} \qquad (2)$$

Покажем, что E – сильное бимоделирование на L. Отсюда будет следовать $s_1 \sim s_2$.

Докажем, что для E выполнено первое условие из определения 11. Пусть $(t_1, t_2) \in E$ и $t_1 \xrightarrow{\alpha} t_1'$. Нам нужно доказать существование такого $t_2' \in S$, что $t_2 \xrightarrow{\alpha} t_2'$ и $(t_1', t_2') \in E$. Предположим, что первое условие из определения 11 не выполнено и такого t_2' не существует. Рассмотрим множество $\{u \mid t_2 \xrightarrow{\alpha} u\}$. Так как L конечно-ветвящаяся, это множество конечно. Пусть все его элементы есть $u_1, \ldots, u_m$. По построению множества $\{u_i\}$ имеем $(t_1', u_i) \notin E$ для любого i. По определению E это значит, что для любого i существует формула f_i, такая что

- $t_1' \models f_i$ и $u_i \not\models f_i$, либо
- $t_1' \not\models f_i$ и $u_i \models f_i$.

Если $t_1' \not\models f_i$, то формулу f_i заменим на ее отрицание. После этой замены получим, что $t_1' \models f_i$ и $u_i \not\models f_i$ для любого i. Рассмотрим формулу $f = \langle\alpha\rangle \bigwedge_{i=1}^{m} f_i$. Эта формула истинна в состоянии t_1, но ложна в состоянии t_2. Мы получили противоречие с тем, что $(t_1, t_2) \in E$, предположив, что первое условие из определения 11 не выполнено. Значит, оно выполнено. Выполнение второго условия доказывается аналогично. Итак, мы доказали, что E – сильное бимоделирование на L, а значит $s_1 \sim s_2$. □

Пример 2 Применим данную теорему для доказательства неэквивалентности графов на рисунке 1. Рассмотрим формулу $f = \langle a\rangle(\langle b\rangle \top \wedge \langle c\rangle \top)$. Эта формула истинна в начальном состоянии графа G_1 и ложна в начальном состоянии графа G_2. Следовательно, эти графы не являются сильно эквивалентными. □

2.6 Алгоритм построения сильной эквивалентности

В этом разделе мы рассмотрим алгоритм, строящий отношение $\sim_L$ сильной эквивалентности состояний конечной системы переходов $L = (S, R)$. Для нахождения $\sim_L$ мы построим монотонно убывающую последовательность бинарных отношений

$$\sim_0 \supseteq \sim_1 \supseteq \ldots \supseteq \sim_n \supseteq \ldots, \qquad \text{где} \quad \forall i \, : \sim_i \subseteq S^2$$

Так как множество S конечно, любая такая последовательность стабилизируется, то есть $\sim_n = \sim_{n+1} = \ldots$ для некоторого n. Отношения $\sim_i$ мы построим так, что отношение $\sim_L$ совпадает с $\sim_n$. Первым элементом последовательности является полное отношение на множестве S: $\sim_0 \stackrel{def}{=} S^2$. Для построения остальных отношений $\sim_i$ мы определим функцию $\mathcal{F} : \mathcal{P}(S^2) \to \mathcal{P}(S^2)$. Она ставит в соответствие бинарному отношению $E \subseteq S^2$ отношение $\mathcal{F}(E) \subseteq S^2$. Последовательность $\{\sim_i\}$ будет задаваться соотношением $\sim_{i+1} \stackrel{def}{=} \mathcal{F}(\sim_i)$.

Определение 16 (Функция $\mathcal{F}$) Пусть $E \subseteq S^2$ и $s_1, s_2 \in S$ – два состояния. Пара (s_1, s_2) принадлежит отношению $\mathcal{F}(E)$ тогда и только тогда, когда

1. из $s_1 \xrightarrow{\alpha} s_1'$ следует существование такого $s_2' \in S$, что $s_2 \xrightarrow{\alpha} s_2'$ и $(s_1', s_2') \in E$;
2. из $s_2 \xrightarrow{\alpha} s_2'$ следует существование такого $s_1' \in S$, что $s_1 \xrightarrow{\alpha} s_1'$ и $(s_1', s_2') \in E$. □

В определении функции $\mathcal{F}$ использованы те же условия 1 и 2, что и в определении сильного бимоделирования 11. Верно следующее утверждение[6]:

Утверждение 4 *Пусть $L = (S, R)$ – размеченная система переходов, а E – бинарное отношение на S. Отношение E является сильным бимоделированием на L тогда и только тогда, когда $E \subseteq \mathcal{F}(E)$.* □

Важным свойством функции $\mathcal{F}$ является ее *монотонность*:

Утверждение 5 *Функция $\mathcal{F}$ монотонна, то есть для любых двух отношений E_1 и E_2 из того, что $E_1 \subseteq E_2$, следует, что $\mathcal{F}(E_1) \subseteq \mathcal{F}(E_2)$.* □

Из монотонности $\mathcal{F}$ следует монотонность последовательности $\{\sim_i\}$.

Утверждение 6 *Последовательность $\{\sim_i\}$ является монотонно нестрого убывающей, то есть для любого $i \geq 0$ верно $\sim_{i+1} \subseteq \sim_i$.*

Доказательство. Докажем данное утверждение индукцией по i. Для $i = 0$ оно верно, так как $\sim_0 = S^2$ и $\sim_1 \subseteq S^2$. Предположим теперь, что $\sim_i \subseteq \sim_{i-1}$ для некоторого $i > 0$. Из монотонности функции $\mathcal{F}$ следует $\mathcal{F}(\sim_i) \subseteq \mathcal{F}(\sim_{i-1})$. Так как по определению $\mathcal{F}(\sim_i) = \sim_{i+1}$ и $\mathcal{F}(\sim_{i-1}) = \sim_i$, отсюда следует $\sim_{i+1} \subseteq \sim_i$. Таким образом, шаг индукции сделан, и утверждение доказано. □

Как упоминалось выше $\sim_n = \sim_{n+1}$ для некоторого n. Обозначим отношение $\sim_n$ через $\sim_*$. Покажем, что отношение $\sim_*$ совпадает с отношением $\sim_L$:

Теорема 7 *Пусть $L = (S, R)$ – размеченная система переходов. Пусть множества S и R конечны. Если отношение $\sim_*$ определено, как описано выше, то $\sim_* = \sim_L$.*

Доказательство. Покажем вначале, что $\sim_*$ – сильное бимоделирование. По построению, $\mathcal{F}(\sim_*) = \sim_{n+1} = \sim_n = \sim_*$. Значит, $\sim_* \subseteq \mathcal{F}(\sim_*)$ и по утверждению 4 $\sim_*$ является сильным бимоделированием.

Покажем теперь, что $\sim_*$ – максимальное сильное бимоделирование на L, то есть $E \subseteq \sim_*$ для любого сильного бимоделирования E на L. Пусть E – произвольное сильное бимоделирование на L. Покажем, что $E \subseteq \sim_i$ для любого i. Отсюда будет следовать, что $E \subseteq \sim_n = \sim_*$. Доказательство проведем индукцией по i.

[6] В книге [Mil89] на странице 104 функция, подобная $\mathcal{F}$, вводится для термов CCS. Там же сформулированы утверждения, аналогичные утверждениям 4 и 5.

Для $i = 0$ доказываемое верно, так как $\sim_0 = S^2$, и $E \subseteq S^2$. Предположим теперь, что $E \subseteq \sim_{i-1}$ для некоторого $i > 0$. Используя утверждения 4 и 5, мы можем записать

$$E \subseteq \mathcal{F}(E) \subseteq \mathcal{F}(\sim_{i-1}) = \sim_i$$

Таким образом, мы сделали шаг индукции и получили требумое утверждение для любого i.

Итак, отношение $\sim_*$ есть максимальное сильное бимоделирование на L. По утверждению 2 оно совпадает с $\sim_L$. $\square$

Данная теорема позволяет построить алгоритм, определяющий эквивалентность двух состояний s_1 и s_2 в некоторой системе переходов. Для решения этой задачи сначала можно построить отношение $\sim_L$ сильной эквивалентности состояний L, а затем проверить, принадлежит ли пара (s_1, s_2) отношению $\sim_L$.

2.7 Минимизация процессных графов

Рассмотрим теперь следующую задачу. Дан конечный процессный граф $G = (S, s_0, R)$. Необходимо построить граф $G_{min} = (S^*, s_0^*, R^*)$, который является минимальным по количеству состояний графом, сильно эквивалентным G. То есть

$$G_{min} \sim G \qquad \text{и} \qquad |S^*| = \min \left\{ |S'| \,\middle|\, (S', s_0', R') \sim G \right\}$$

Во-первых, отметим, что если в графе G есть недостижимые состояния, то, удалив их, мы получим граф, сильно эквивалентный G. Следующее утверждение доказывает, что это так:

Утверждение 8 *Пусть граф G' получается из графа G удалением и добавлением недостижимых состояний. Тогда $G' \sim G$.* $\square$

Доказательство этого утверждения остается читателю в качестве упражнения. Удаляя недостижимые состояния, не всегда можно получить минимальный граф, сильно эквивалентный исходному. Например, граф G_1' на рисунке 2 не имеет недостижимых состояний, а минимальный граф, сильно эквивалентный ему есть $a.b.NIL$. Рассмотрим другой метод минимизации графов, основанный на построении фактор-множества множества состояний графа по некоторому отношению эквивалентности.

Определение 17 Пусть $G = (S, s_0, R)$ – процессный граф, а E – некоторое отношение эквивалентности на множестве S. ***Фактор-графом*** графа G по отношению E мы будем называть граф $G/_E = (S/_E, [s_0]_E, R/_E)$. Он имеет следующие компоненты:

- Множество состояний графа $G/_E$ есть фактор-множество $S/_E$. Класс эквивалентности элемента $s \in S$ мы будем обозначать записью $[s]_E$.

- Класс эквивалентности $[s_0]_E$ начального состояния является начальным состоянием фактор-графа.

- $[s_1]_E \xrightarrow{\alpha} [s_2]_E$ в $G/_E \stackrel{def}{\Longleftrightarrow} \exists s_1' \in [s_1]_E, s_2' \in [s_2]_E \,:\, s_1' \xrightarrow{\alpha} s_2'$ в G. $\square$

Утверждение 9 *Пусть $G = (S, s_0, R)$ – процессный граф, а отношение E – сильное бимоделирование на (S, R). Тогда $G/_E \sim G$.*

Доказательство. Не ограничивая общности, можно считать, что графы G и $G/_E$ не содержат общих состояний. Определим отношение $I \subseteq \left(S \cup S/_E\right)^2$ следующим образом:

$$I = \{(s, [s]_E) \mid s \in S\}$$

Покажем, что I – сильное бимоделирование на РСП $(S \cup S/_E, R \cup R/_E)$. Проверим для I условия из определения 11.

Если $\alpha \in Act$ и $s \xrightarrow{\alpha} t$, то $[s]_E \xrightarrow{\alpha} [t]_E$ по определению отношения $R/_E$. Первое условие, таким образом, выполнено.

Пусть теперь $[s]_E \xrightarrow{\alpha} [t]_E$. Укажем такое состояние u, что $s \xrightarrow{\alpha} u$ и $(u, [t]_E) \in I$. По определению $R/_E$ существуют такие состояния s' и t', что $(s, s') \in E$, $s' \xrightarrow{\alpha} t'$ и $(t, t') \in E$. Так как E – сильное бимоделирование, существует такое состояние u, что $s \xrightarrow{a} u$ и $(u, t') \in E$. Из транзитивности E следует, что $(u, t) \in E$ и, следовательно, $[u]_E = [t]_E$. Значит, $(u, [t]_E) \in I$ по определению I. Таким образом, u является подходящим состоянием.

Итак, I – сильное бимоделирование. Так как $(s_0, [s_0]_E) \in I$, графы G и $G/_E$ сильно эквивалентны. □

Если граф G не содержит недостижимых состояний, то граф $G/_{\sim_G}$ является минимальным графом, сильно эквивалентным G. Для доказательства этого нам потребуются две леммы.

Лемма 10 *Пусть даны*

- *размеченная система переходов $L = (S, R)$,*
- *непустое множество Q, не пересекающееся с S (то есть $Q \cap S = \emptyset$),*
- *множество переходов $T \subseteq Q \times Act \times (S \cup Q)$.*

Пусть M есть система переходов $(S \cup Q, R \cup T)$. Тогда $\sim_L = \sim_M \cap S^2$.

Иными словами, пусть система переходов M образуется из системы переходов L

- добавлением новых состояний (образующих множество Q),
- добавлением множества T переходов из новых состояний.

Тогда любые два состояния из L сильно эквивалентны в L тогда и только тогда, когда они сильно эквивалентны в M.

Доказательство. Для доказательства этой леммы вначале заметим, что

1. $\sim_L$ является сильным бимоделированием на M,
2. $\sim_M \cap S^2$ является сильным бимоделированием на L.

Доказательства этих двух утверждений несложны и остаются читателю в качестве упражнений. Из пункта 1 следует, что $\sim_L \subseteq \sim_M$. Так как $\sim_L \subseteq S^2$, выполнено $\sim_L \subseteq \sim_M \cap S^2$. Из пункта 2 вытекает $\sim_M \cap S^2 \subseteq \sim_L$. В итоге получаем $\sim_L = \sim_M \cap S^2$. □

Лемма 11 *Пусть* $G_1 = (S^{(1)}, s_0^{(1)}, R^{(1)})$ *и* $G_2 = (S^{(2)}, s_0^{(2)}, R^{(2)})$ *– два процессных графа, множества состояний которых не пересекаются:* $S^{(1)} \cap S^{(2)} = \emptyset$*. Пусть* $G_1 \sim G_2$ *и состояние* $s_1 \in S^{(1)}$ *достижимо в* G_1*. Тогда существует такое состояние* s_2 *в* G_2*, что* $s_1 \sim s_2$ *в системе переходов* $(S^{(1)} \cup S^{(2)}, R^{(1)} \cup R^{(2)})$*.*

Доказательство. Так как s_1 достижимо в G_1, существует путь

$$t_1 \xrightarrow{\alpha_1} t_2 \xrightarrow{\alpha_2} \ldots \xrightarrow{\alpha_{n-1}} t_n,$$

где $t_1 = s_0^{(1)}$, $t_n = s_1$ и $n \geq 1$. Индукцией по i покажем, что для любого i, $1 \leq i \leq n$ существует такое состояние $u_i \in S^{(2)}$, что $t_i \sim u_i$.

Если $i = 1$, то в качестве u_i можно взять $s_0^{(2)}$. Если $i > 1$, то по предположению индукции существует такое состояние $u_{i-1} \in S^{(2)}$, что $t_{i-1} \sim u_{i-1}$. Так как $t_{i-1} \xrightarrow{\alpha_{i-1}} t_i$, и $\sim$ является сильным бимоделированием, существует такое состояние u_i, что $u_{i-1} \xrightarrow{\alpha_{i-1}} u_i$ и $t_i \sim u_i$. Так как $u_{i-1} \in S^{(2)}$, u_i также принадлежит $S^{(2)}$. Шаг индукции сделан, и доказательство по индукции завершено. В качестве искомого состояния s_2 мы можем теперь взять u_n. □

Теорема 12 *Пусть* $G = (S, s_0, R)$ *– процессный граф без недостижимых состояний. Пусть* $G^* = G/_{\sim_G} = (S^*, s_0^*, R^*)$*. Пусть граф* $H = (Q, q_0, T)$ *сильно эквивалентен* G*. Тогда* $|S^*| \leq |Q|$*.*

Доказательство. Не ограничивая общности рассуждений, мы можем предположить, что $Q \cap S = \emptyset$. Пусть M есть система переходов $(Q \cup S, R \cup T)$. По лемме 10, $\sim_G = \sim_M \cap S^2$. Далее мы будем обозначать отношения $\sim_G$ и $\sim_M$ одним и тем же символом $\sim$.

Определим функцию $h : S^* \to Q$ следующим образом. Пусть $s \in S$. В классе $[s]_\sim \in S^*$ выберем произвольный элемент s'. Так как s' достижимо в S, и $G \sim H$, по лемме 11 существует такое $q \in Q$, что $q \sim s'$. Значение $h([s]_\sim)$ положим равным произвольному такому q.

Покажем, что построенное отображение h инъективно, то есть из $h([s]_\sim) = h([t]_\sim)$ следует $[s]_\sim = [t]_\sim$. Пусть $h([s]_\sim) = h([t]_\sim) = q$. По определению h для некоторых s' и t' выполнено соотношение $s \sim s' \sim q \sim t' \sim t$. Отсюда следует $s \sim t$ и $[s]_\sim = [t]_\sim$.

Из инъективности h следует $|S^*| \leq |Q|$. □

Данная теорема дает алгоритм построения минимального графа, сильно эквивалентного данному графу $G = (S, s_0, R)$:

1. Сначала в графе G надо удалить все недостижимые состояния.

2. Затем нужно построить отношение сильной эквивалентности $\sim$ на (S, R), как описано в предыдущем разделе.

3. Результатом работы алгоритма является фактор-граф $G/_\sim$.

3 Исчисление взаимодействующих систем (CCS)

3.1 Синтаксис CCS

Мы переходим к изучению исчисления взаимодействующих систем (CCS), которое было предложено Р. Милнером [Mil80]. В CCS мы, как правило, не будем явно задавать процессные графы, а перейдем к к описанию процессов в виде процессных выражений. Множество *процессных выражений* мы обозначим через $\mathcal{E}$. Как и ранее будем предполагать, что задано множество имен действий $\mathcal{N}$, а множество действий Act определяется так же, как на странице 3. Пусть также заданы множества *процессных констант* $\mathcal{K}$ и *процессных переменных* $\mathcal{X}$.

Определение 18 Множество *процессных выражений* $\mathcal{E}$ по определению есть минимальное множество, удовлетворяющее следующим свойствам:

1. Любая процессная константа и любая процессная переменная являются выражениями: $\mathcal{K} \subseteq \mathcal{E}$, $\mathcal{X} \subseteq \mathcal{E}$.

2. Для любого множества индексов I и семейства $\{E_i \mid i \in I\}$ выражений, индексированных элементами I, выражением будет также запись $\sum_{i \in I} E_i$. Такое выражение называется *суммой* или *альтернативной композицией* выражений $\{E_i\}$.

3. Для любых двух выражений $E, F \in \mathcal{E}$, действия $\alpha \in Act$, множества имен $L \subseteq \mathcal{N}$ и функции переименования $f : \mathcal{L} \to \mathcal{L}$ (см. страницу 4) процессными выражениями будут также следующие записи:

   - $\alpha.E$ (*добавление префикса* α к E),
   - $E|F$ (*параллельная композиция* E и F),
   - $E \setminus L$ (*ограничение* E по L) и
   - $E[f]$ (*переименование* E согласно f).

Процессом называется выражение, не содержащее процессных переменных. Множество процессов мы обозначим через $\mathcal{P}$. Мы будем предполагать, что для каждой процессной константы $A \in \mathcal{K}$ задано ее *определение*, имеющее вид $A \stackrel{def}{=} P$, где $P \in \mathcal{P}$. □

Если в сумме $\sum_{i \in I} E_i$ множество I пусто, то такую сумму мы будем обозначать символом $\mathbf{0}$. Когда $I = \{1, 2\}$, сумму $\sum_{i \in I} E_i$ мы будем часто записывать при помощи обычного символа $+$, то есть как $E_1 + E_2$. Для выражений вида $E[f]$ мы будем использовать альтернативную запись $E[\beta_1/a_1, \ldots, \beta_n/a_n]$ аналогично тому, как мы это делали для процессных графов (см. страницу 4). Если $L \subseteq \mathcal{L}$, то запись $\bar{L}$ будет обозначать множество $\{\bar{l} \mid l \in L\}$.

Для произвольных процессных выражений E и F выражение $E\{F/X\}$ обозначает выражение, получающийся в результате подстановки F вместо всех вхождений переменной X в выражение E. Если $F_1, \ldots, F_n$ есть последовательность выражений, а $X_1, \ldots, X_n$ – последовательность различных переменных, то мы будем использовать выражение $E\{F_1/X_1, \ldots, F_n/X_n\}$ как альтернативную запись выражения $E\{F_1/X_n\} \cdots \{F_n/X_n\}$. Для процессного выражения E запись $\mathrm{vars}\,(E)$ будет обозначать множество всех процессных переменных, входящих в E. Процессные переменные понадобятся нам в разделе 3.4, а до этого раздела будут рассматриваться только выражения без переменных, то есть процессы.

Каждый процесс из множества $\mathcal{P}$ можно использовать для описания поведения некоторой системы в определенном состоянии. При этом поведение одной и той же системы в разных состояниях может описываться разными процессами. На процессах мы определим отношение перехода. Если P и Q – процессы, а α – действие, то запись $P \xrightarrow{\alpha} Q$ может означать следующее. Если поведение некоторой системы описывается процессом P, то система может выполнить действие α так, что после этого ее поведение будет описываться процессом[7] Q. В этом случае мы будем также говорить, что существует переход из процесса P к процессу Q с выполнением действия α.

Отношение перехода на процессах мы зададим аксиоматически, при помощи правил вывода. Каждое правило вывода

- может содержать несколько *посылок* (возможно, не содержит ни одной),
- содержит *заключение*,
- может иметь *условие применимости*.

Все посылки и заключение являются обозначениями перехода в виде $P \xrightarrow{\alpha} Q$. Правило вывода (если у него есть условие применимости) записывается в следующем виде:

$$\frac{\text{посылка}_1 \quad \dots \quad \text{посылка}_n}{\text{заключение}} \quad (\text{условие применимости})$$

Здесь $n \geq 0$. Такое правило позволяет сделать вывод о существовании перехода, стоящего в заключении правила, если

- существуют переходы, являющиеся посылками правила, и
- выполнено условие применимости.

Аксиомы и правила вывода для отношения перехода изображены на рисунке 3.

Определение 19 *Выводом* перехода $P \xrightarrow{\alpha} Q$ называется такая последовательность переходов $P_1 \xrightarrow{\alpha_1} Q_1, \dots, P_n \xrightarrow{\alpha_n} Q_n$, что

- $P_n = P$, $Q_n = Q$, $\alpha_n = \alpha$.
- Для любого i можно доказать существование перехода $P_i \xrightarrow{\alpha_i} Q_i$ применением некоторого правила вывода. При этом в качестве посылок правила должны быть только переходы, находящиеся в выводе перед i-м переходом.

Переход $P \xrightarrow{\alpha} Q$ называется *выводимым*, если для него существует вывод. Мы будем считать, что существует переход из P в Q с выполнением действия α, если переход $P \xrightarrow{\alpha} Q$ выводим. □

$$\textbf{Act}\quad \frac{}{\alpha.P \xrightarrow{\alpha} P} \qquad\qquad \textbf{Con}\quad \frac{P \xrightarrow{\alpha} P'}{A \xrightarrow{\alpha} P'} \quad (A \stackrel{def}{=} P)$$

$$\textbf{Sum}_j\quad \frac{P_i \xrightarrow{\alpha} P'}{\sum\limits_{i\in I} P_i \xrightarrow{\alpha} P'} \quad (j \in I)$$

$$\textbf{Com}_1\quad \frac{P \xrightarrow{\alpha} P'}{P|Q \xrightarrow{\alpha} P'|Q} \qquad\qquad \textbf{Com}_2\quad \frac{Q \xrightarrow{\alpha} Q'}{P|Q \xrightarrow{\alpha} P|Q'}$$

$$\textbf{Com}_3\quad \frac{P \xrightarrow{\alpha} P' \quad Q \xrightarrow{\bar{\alpha}} Q'}{P|Q \xrightarrow{\tau} P'|Q'} \quad (\alpha \neq \tau)$$

$$\textbf{Res}\quad \frac{P \xrightarrow{\alpha} P'}{P \setminus L \xrightarrow{\alpha} P'} \quad (\alpha, \bar{\alpha} \notin L) \qquad\qquad \textbf{Rel}\quad \frac{P \xrightarrow{\alpha} P'}{P[f] \xrightarrow{f(\alpha)} P'[f]}$$

Рис. 3: Правила вывода для отношения перехода

Правило **Act** не имеет посылок. Оно позволяет вывести переход $\alpha.P \xrightarrow{\alpha} P$ для любого действия α и процесса P. Для процесса $\mathbf{0}$ правил вывода нет. Этот процесс описывает поведение *неактивной* системы, то есть системы, которая не может выполнить ни одного действия. Согласно правилам $\mathbf{Sum}_j$ переход $\sum_{i\in I} P_i \xrightarrow{\alpha} P'$ существует, если есть переход $P_j \xrightarrow{\alpha} P'$ для некоторого $j \in I$. Этот индекс j добавляется к слову **Sum**, обозначающему правило. Для частного вида сумм $P_1 + P_2$ мы получаем два правила вывода (для $j = 1$ и $j = 2$):

$$\textbf{Sum}_1\quad \frac{P_1 \xrightarrow{\alpha} P'}{P_1 + P_2 \xrightarrow{\alpha} P'} \qquad\qquad \textbf{Sum}_2\quad \frac{P_2 \xrightarrow{\alpha} P'}{P_1 + P_2 \xrightarrow{\alpha} P'}$$

Пример 3 Для примера докажем существование перехода

$$((\bar{a}.\mathbf{0} + \bar{b}.\mathbf{0})|a.\mathbf{0}) \setminus \{a\} \xrightarrow{\tau} (\mathbf{0}|\mathbf{0}) \setminus \{a\}$$

Он выводится в результате следующих применений правил вывода:

(1): $\bar{a}.\mathbf{0} \xrightarrow{a} \mathbf{0}$, по **Act**
(2): $\bar{a}.\mathbf{0} + \bar{b}.\mathbf{0} \xrightarrow{a} \mathbf{0}$, из (1) по $\mathbf{Sum}_1$
(3): $a.\mathbf{0} \xrightarrow{a} \mathbf{0}$, по **Act**
(4): $(\bar{a}.\mathbf{0} + \bar{b}.\mathbf{0})|a.\mathbf{0} \xrightarrow{\tau} \mathbf{0}|\mathbf{0}$, из (2) и (3) по $\mathbf{Com}_3$
(5): $((\bar{a}.\mathbf{0} + \bar{b}.\mathbf{0})|a.\mathbf{0}) \setminus \{a\} \xrightarrow{\tau} (\mathbf{0}|\mathbf{0}) \setminus \{a\}$, из (4) по **Res**

□

[7] Часто термин "процесс" в этом пособии будет означать не только выражение, но и тот объект, который это выражение описывает. Так, например, если $P \xrightarrow{\alpha} Q$, мы будем говорить, что процесс P выполняет действие α и далее ведет себя как Q.

Множество $\mathcal{P}$ вместе с отношением перехода образует размеченную систему переходов. Процессы является в ней состояниями, а множество переходов есть

$$\mathcal{R} \stackrel{def}{=} \{(P, \alpha, P') \mid P \stackrel{\alpha}{\longrightarrow} P'\}$$

Сильная эквивалентность процессов определяется как эквивалентность процессов в этой РСП.

Определение 20 Процессы P и Q сильно эквивалентны тогда и только тогда, когда $P \sim Q$ в РСП $(\mathcal{P}, \mathcal{R})$. □

Мы будем говорить, что процесс Q *достижим* из процесса P, если существуют такие последовательность процессов $P_1, \ldots, P_n$ и последовательность действий $\alpha_1, \ldots, \alpha_{n-1}$, что $n \geq 1$, $P_1 = P$, $P_n = Q$, и для любого i, $1 \leq i < n$, существует переход $P_i \stackrel{\alpha_i}{\longrightarrow} P_{i+1}$.

Определение 21 Пусть P – процесс. Граф переходов P – это процессный граф $TG(P)$, который является результатом удаления недостижимых состояний в графе $(\mathcal{P}, P, \mathcal{R})$. □

Пример 4 Применяя процессные константы, можно задавать графы с циклами. Пусть, для примера, у нас заданы две процессные константы A и B, рекурсивные определения которых суть

$$A \stackrel{def}{=} a.B \qquad\qquad B \stackrel{def}{=} b.A$$

Из правил **Act** и **Def** следует, что для этих констант выполнено $A \stackrel{a}{\longrightarrow} B$ и $B \stackrel{b}{\longrightarrow} A$. Для примера покажем, что существует переход $A \stackrel{a}{\longrightarrow} B$.

(1): $a.B \stackrel{a}{\longrightarrow} B$, по **Act**

(2): $A \stackrel{a}{\longrightarrow} B$, из (1) по **Def**, так как $A \stackrel{def}{=} a.B$

□

3.2 Сорта процессных выражений

Зададим для каждой константы $A \in \mathcal{K}$ некоторое множество $\mathcal{L}(A) \subseteq \mathcal{L}$, и для каждой процессной переменной $X \in \mathcal{X}$ – некоторое множество $\mathcal{L}(X) \subseteq \mathcal{L}$. Множество $\mathcal{L}(A)$ или $\mathcal{L}(X)$ мы будем называть *сортом* константы A или переменной X соответственно.

Определение 22 ([Mil89], стр. 52) Пусть заданы сорта процессных констант и процессных переменных. *Сортом* выражения E мы будем называть множество $\mathcal{L}(E)$, вычисляемое по следующим правилам:

$$\begin{array}{rclrcl}
\mathcal{L}(\mathbf{0}) & = & \emptyset & \mathcal{L}(\sum_{i \in I} E_i) & = & \bigcup_{i \in I} \mathcal{L}(E_i) \\
\mathcal{L}(E|F) & = & \mathcal{L}(E) \cup \mathcal{L}(F) & \mathcal{L}(\alpha.E) & = & \{\alpha\} \cup \mathcal{L}(E), \quad \alpha \neq \tau \\
\mathcal{L}(\tau.E) & = & \mathcal{L}(E) & \mathcal{L}(E \setminus L) & = & \mathcal{L}(E) \setminus (L \cup \bar{L}) \\
\mathcal{L}(E[f]) & = & f(\mathcal{L}(E)) & & &
\end{array}$$

Для множества S выражение $f(S)$ здесь обозначает множество

$$\{f(s) \mid s \in S\}$$

$\square$

При задании рекурсивных определений $A \stackrel{def}{=} P$ для процессных констант мы всегда будем требовать, чтобы выполнялось соотношение

$$\mathcal{L}(A) \supseteq \mathcal{L}(P) \tag{3}$$

Верно следующее утверждение:

Утверждение 13 ([Mil89]) *Пусть $P \xrightarrow{\alpha} P'$. Тогда $\alpha \in \mathcal{L}(P)$ и $\mathcal{L}(P') \subseteq \mathcal{L}(P)$.* $\square$

Доказательство. Доказательство проведем индукцией по минимальной длине вывода для перехода $P \xrightarrow{\alpha} P'$. Рассмотрим кратчайший вывод этого перехода. Рассуждения будут различны в зависимости от того, при помощи какого правила был получен переход $P \xrightarrow{\alpha} P'$ в этом выводе.

1. Если данный переход был выведен при помощи правила **Act**, то $P' = \alpha.P$. Доказываемое утверждение следует из определения $\mathcal{L}(\alpha.P)$.

2. Пусть переход был выведен при помощи правила $\mathbf{Sum}_j$. Тогда $P = \sum_{i \in I} P_i$ и $P_j \xrightarrow{\alpha} P'$. К переходу $P_j \xrightarrow{\alpha} P'$ можно применить предположение индукции, так как минимальная длина вывода для него меньше, чем для перехода $P \xrightarrow{\alpha} P'$. По предположению индукции, получаем $\alpha \in \mathcal{L}(P_j)$ и $\mathcal{L}(P') \subseteq \mathcal{L}(P_j)$. Доказываемое утверждение следует теперь из определения $\mathcal{L}(\sum_{i \in I} P_i)$.

3. Для остальных вариантов последнего шага вывода доказательство проводится аналогично. $\square$

3.3 Эквивалентные преобразования процессов

Теперь давайте рассмотрим ряд полезных соотношений между процессами, при помощи которых можно осуществлять их эквивалентные преобразования. Нам потребуется следующее свойство[8] отношения $\sim$:

Утверждение 14 *$P \sim Q$ тогда и только тогда, когда выполнены следующие условия:*

1. *Из $P \xrightarrow{\alpha} P'$ следует существование такого Q', что $Q \xrightarrow{a} Q'$ и $P' \sim Q'$;*

2. *Из $Q \xrightarrow{\alpha} Q'$ следует существование такого P', что $P \xrightarrow{\alpha} P'$ и $P' \sim Q'$.*

[8] Доказательство этого утверждения, использованное в настоящем пособии, приведено в работе [Mil90] на странице 1217.

Доказательство. Для доказательства утверждения достаточно показать, что $\sim = \mathcal{F}(\sim)$. Функция $\mathcal{F}$ была введена на странице 11. Так как $\sim$ – сильное бимоделирование, по утверждению 4 выполнено включение $\sim \subseteq \mathcal{F}(\sim)$. По утверждению 5 функция $\mathcal{F}$ мотононна. Следовательно, $\mathcal{F}(\sim) \subseteq \mathcal{F}\big(\mathcal{F}(\sim)\big)$. По утверждению 4 отсюда следует, что $\mathcal{F}(\sim)$ является сильным бимоделированием. Так как $\sim$ – максимальное сильное бимоделирование, выполнено включение $\mathcal{F}(\sim) \subseteq \sim$. Мы доказали, что каждое из отношений $\sim$ и $\mathcal{F}(\sim)$ является подмножеством другого. Поэтому $\sim = \mathcal{F}(\sim)$. □

Перейдем теперь к соотношениям для процессов.

Утверждение 15 (законы моноида, [Mil89], стр. 94) *1. $P+Q \sim Q+P$;*

2. $P+(Q+R) \sim (P+Q)+R$;

3. $P+P \sim P$;

4. $P+\mathbf{0} \sim P$.

Доказательство. Для примера докажем пункт (1). Остальные пункты доказываются аналогичными рассуждениями. Воспользуемся утверждением 14. Пусть существует переход $P+Q \xrightarrow{\alpha} R$. Возможны два варианта:

1. Он был выведен по правилу $\mathbf{Sum}_1$ (и тогда $P \xrightarrow{\alpha} R$).

2. Он был выведен по по правилу $\mathbf{Sum}_2$ (и тогда $Q \xrightarrow{\alpha} R$).

В обоих случаях можно вывести переход $Q+P \xrightarrow{\alpha} R$: в первом случае – по правилу $\mathbf{Sum}_2$, а во втором – по правилу $\mathbf{Sum}_1$. Так как $R \sim R$ для любого R, мы получили, что выполнены условия утверждения 14 для процессов $P+Q$ и $Q+P$. Следовательно, $P+Q \sim Q+P$. □

Утверждение 16 (статические законы, [Mil89], стр. 94) *1. $P|Q \sim Q|P$;*

2. $P|(Q|R) \sim (P|Q)|R$;

3. $P|\mathbf{0} \sim P$;

4. $P \setminus L \sim P$, если $\mathcal{L}(P) \cap (L \cup \bar{L}) = \emptyset$;

5. $(P \setminus K) \setminus L \sim P \setminus (K \cup L)$;

6. $(P[f]) \setminus L \sim (P \setminus M)[f]$, где $M = \big\{a \in \mathcal{N} \mid f(a)$ или $\overline{f(a)} \in L\big\}$;

7. $(P|Q) \setminus L \sim (P \setminus L)|(Q \setminus L)$, если $\mathcal{L}(P) \cap \overline{\mathcal{L}(Q)} \cap (L \cup \bar{L}) = \emptyset$;

8. $P[Id_{Act}] \sim P$;

9. $P[f] \sim P[f']$, если $f|_{\mathcal{L}(P)} = f'|_{\mathcal{L}(P)}$;

10. $P[f][f'] \sim P[f' \circ f]$;

11. *$(P|Q)[f] \sim (P[f])|(Q[f])$, если f не отождествляет никакие два элемента из множества $L \cup \bar{L}$, где $L = \mathcal{L}(P|Q) \cup \overline{\mathcal{L}(P|Q)}$. Последнее условие значит, что если $a, b \in L \cup \bar{L}$ и $a \neq b$, то $f(a) \neq f(b)$.*

Доказательство. Для примера докажем пункт (1). Остальные пункты доказываются аналогичными рассуждениями. Определим отношение E на множестве процессов как $E = \{(P|Q, Q|P) \mid P, Q \in \mathcal{P}\}$. Докажем, что E – сильное бимоделирование. Покажем, что выполнено первое условие из определения 11 сильного бимоделирования. Пусть существует переход $P|Q \xrightarrow{\alpha} R$. Возможны три варианта:

1. Он был выведен по правилу $\mathbf{Com}_1$ и $R = P'|Q$, где $P \xrightarrow{\alpha} P'$. Тогда по правилу $\mathbf{Com}_2$ получаем $Q|P \xrightarrow{\alpha} Q|P'$, причем $(P'|Q, Q|P') \in E$.

2. Если переход был выведен по правилу $\mathbf{Com}_2$, то рассуждения аналогичны.

3. Если переход был выведен по правилу $\mathbf{Com}_3$, то $\alpha \neq \tau$ и $R = P'|Q'$. При этом процессы P' и Q' таковы, что $P \xrightarrow{\beta} P'$ и $Q \xrightarrow{\bar{\beta}} Q'$ для некоторого $\beta \neq \tau$. По правилу $\mathbf{Com}_3$ в этом случае получаем $Q|P \xrightarrow{\tau} Q'|P'$, причем $(P'|Q', Q'|P') \in E$.

Тем самым первое условие определения 11 доказано. Доказав симметричными рассуждениями второе условие, получим, что E – сильное бимоделирование. Отсюда следует сильная эквивалентность $P|Q \sim Q|P$ для произвольных процессов P и Q. □

Утверждение 17 *Пусть $A \in \mathcal{K}$, $P \in \mathcal{P}$ и $A \stackrel{def}{=} P$. Тогда $A \sim P$.* □

Последнее утверждение доказывается при помощи утверждения 14.

3.4 Сохранение сильной эквивалентности операциями построения процессов

Определение 23 Пусть A – произвольное множество. *Операцией* на множестве A мы будем называть отображение $f : A^n \to A$, где $n \geq 0$. Число n называется *арностью* операции f. □

Множество A^0 по определению состоит из одного элемента для любого A. Операция арности 0 – это отображение $f : A^0 \to A$. Такая операция однозначно определяется некоторым элементом множества A.

Определение 24 Пусть $f : A^n \to A$ есть операция на множестве A, а $\sim$ – отношение эквивалентности на A. Операция f *сохраняет* отношение $\sim$, если для любых $a_1, \dots, a_n, b_1, \dots, b_n \in A$ из того, что $a_i \sim b_i$ для любого i, следует $f(a_1, \dots, a_n) \sim f(b_1, \dots, b_n)$. □

На множестве процессов $\mathcal{P}$ определим операции "*построения процессов*":

- Пусть $\alpha \in Act$, $L \subseteq \mathcal{N}$, и $f : \mathcal{L} \to \mathcal{L}$ – функция переименования. Тогда "$\alpha.$", "$\backslash L$" и "$[f]$" являются операциями арности один. Они определяются следующим образом:

$$\alpha. : P \mapsto \alpha.P \qquad \setminus L : P \mapsto P \setminus L \qquad [f] : P \mapsto P[f]$$

  для любого $P \in \mathcal{P}$.

- "$+$" и "$|$" есть операции арности два. Они определяются следующими соотношениями:

$$+ : (P, Q) \mapsto P + Q \qquad | : (P, Q) \mapsto P|Q$$

  где $P, Q \in \mathcal{P}$.

Следующая теорема и следствие из нее показывают, что все введенные операции сохраняют отношение сильной эквивалентности. Второй пункт этой теоремы касается сумм с произвольным количеством слагаемых. В частном случае, когда число слагаемых в сумме равно двум, этот пункт утверждает, что операция "$+$" (сумма двух слагаемых) сохраняет отношение $\sim$.

Теорема 18 ([Mil89], стр. 97) *1. Пусть $P, Q, S \in \mathcal{P}$, $\alpha \in Act$, $L \subseteq \mathcal{N}$ и $f : \mathcal{L} \to \mathcal{L}$ – функция переименования. Пусть также $P \sim Q$. Тогда выполнены следующие соотношения:*

(a) $\alpha.P \sim \alpha.Q$,
(b) $P|S \sim Q|S$,
(c) $P \setminus L \sim Q \setminus L$,
(d) $P[f] \sim Q[f]$.

2. Пусть I – некоторое множество и $\{P_i\}_{i \in I}$, $\{Q_i\}_{i \in I}$ – семейства процессов, индексированных элементами I. Пусть также $P_i \sim Q_i$ для любого $i \in I$. Тогда $\sum_{i \in I} P_i \sim \sum_{i \in I} Q_i$.

Доказательство. Докажем вначале второй пункт. Обозначим процесс $\sum_{i \in I} P_i$ через P, а процесс $\sum_{i \in I} Q_i$ – через Q. Воспользуемся утверждением 14. Покажем, что для процессов P и Q выполнены условия этого утверждения. Пусть $P \xrightarrow{\alpha} P'$ для некоторых $\alpha \in Act$ и $P' \in \mathcal{P}$. Этот переход может быть выведен только по правилам вида $\mathbf{Sum}_j$, где $j \in I$. Следовательно, существует такое $j \in I$, что $P_j \xrightarrow{\alpha} P'$. Так как $P_j \sim Q_j$, существует такое $Q' \in \mathcal{P}$, что $Q_j \xrightarrow{\alpha} Q'$ и $P' \sim Q'$. При этом $Q \xrightarrow{\alpha} Q'$.

Итак, первое условие утверждения 14 доказано. Второе условие доказывается аналогично. По утверждению 14 получаем $P \sim Q$.

Пункт 1a теоремы 18 доказывается аналогично пункту 2.

Докажем теперь пункт 1b. Определим отношение $E \subseteq \mathcal{P}^2$ следующим образом:

$$E = \{(P|S, Q|S) \;\mid\; P, Q, R \in \mathcal{P},\; P \sim Q\}$$

Покажем, что E – сильное бимоделирование. Пусть $(P|S, Q|S)$ – произвольная пара, принадлежащая E. По построению E имеем $P \sim Q$. Покажем, что для данной пары выполнены условия из определения сильного бимоделирования (определение 11). Пусть $P|S \xrightarrow{\alpha} R$ для некоторых $\alpha \in Act$ и $R \in \mathcal{P}$. Рассмотрим три случая.

1. Этот переход выводится по правилу $\mathbf{Com}_1$. В этом случае $P \xrightarrow{\alpha} P'$ и $R = P'|S$ для некоторого $P' \in \mathcal{P}$. Так как $P \sim Q$, существует такой процесс Q', что $Q \xrightarrow{\alpha} Q'$ и $P' \sim Q'$. По правилу $\mathbf{Com}_1$ можно вывести переход $Q|S \xrightarrow{\alpha} Q'|S$. При этом $(R, Q'|S) \in E$.

2. Если переход выводится по правилу $\mathbf{Com}_2$, то $S \xrightarrow{\alpha} S'$ и $R = P|S'$ для некоторого $S' \in \mathcal{P}$. В этом случае получаем $Q|S \xrightarrow{\alpha} Q|S'$ и $(R, Q|S') \in E$.

3. Пусть, наконец, вывод осуществляется по правилу $\mathbf{Com}_3$. Тогда $\alpha = \tau$. При этом $P \xrightarrow{\beta} P'$, $S \xrightarrow{\bar{\beta}} S'$ и $R = P'|S'$ для некоторых $P', S' \in \mathcal{P}$ и действия β, не равного τ. Поскольку $P \sim Q$, существует такой $Q' \in \mathcal{P}$, что $Q \xrightarrow{\beta} Q'$ и $P' \sim Q'$. Нетрудно убедиться, что $Q|S \xrightarrow{\tau} Q'|S'$ и $(R, Q'|S') \in E$.

Мы доказали, что для E выполнено первое условие из определения сильного бимоделирования. Доказав аналогично второе условие, мы получим, что E – сильное бимоделирование. Значит, из $P \sim Q$ следует $P|S \sim Q|S$.

Пункты 1c и 1d доказываются аналогично пункту 1b. □

Следствие 19 *Операция "|" сохраняет отношение $\sim$, то есть из $P_1 \sim P_2$ и $Q_1 \sim Q_2$ следует $P_1|Q_1 \sim P_2|Q_2$.*

Доказательство. Пусть $P_1 \sim P_2$ и $Q_1 \sim Q_2$. Воспользуемся пунктом 1b теоремы 18 и коммутативностью операции "|" (пункт 1 утверждения 16).

$P_1\|Q_1 \sim P_2\|Q_1 \sim$	по пункту 1b теоремы 18
$\sim Q_1\|P_2 \sim$	по коммутативности "|"
$\sim Q_2\|P_2 \sim$	по пункту 1b теоремы 18
$\sim P_2\|Q_2$	по коммутативности "|"

□

Следствие 20 *Пусть E – процессное выражение, X – процессная переменная и* $\mathrm{vars}\,(E) \subseteq \{X\}$ *(последнее значит, что E не содержит процессных переменных, кроме, может быть, X). Пусть E содержит ровно одно вхождение X. Пусть также $P, Q \in \mathcal{P}$ и $P \sim Q$. Тогда $E\{P/X\} \sim E\{Q/X\}$.*

Предположим, что E конечно (то есть не содержит сумм с бесконечным числом слагаемых). Тогда доказательство этого следствия можно провести индукцией по длине E. При этом можно применить тот факт, что сильная эквивалентность сохраняется при операциях построения процессов (теорема 18 и следствие 19). Детали этого доказательства а также доказательство в случае бесконечного E в настоящем пособии не приводятся.

Следствие 20 гарантирует, что если заменить в некотором процессе произвольное подвыражение на сильно эквивалентное ему, то процесс до замены будет сильно эквиваленен процессу после замены.

Пример 5 Рассмотрим следующие два процесса:

$$\begin{aligned} P_1 &= arg.\overline{res}.\mathbf{0} \\ P_2 &= arg.\overline{res}.\mathbf{0} + arg.(\overline{res}.\mathbf{0} + \overline{res}.\mathbf{0}) \end{aligned}$$

Покажем, что $P_1 \sim P_2$. Воспользуемся соотношением $P + P \sim P$ (пункт 1 утверждения 15). Из него следует, что $\overline{res}.\mathbf{0} + \overline{res}.\mathbf{0} \sim \overline{res}.\mathbf{0}$. Заменяя в процессе P_2 подвыражение $\overline{res}.\mathbf{0} + \overline{res}.\mathbf{0}$ на $\overline{res}.\mathbf{0}$, получаем

$$P_2 \sim arg.\overline{res}.\mathbf{0} + arg.\overline{res}.\mathbf{0}$$

Применяя снова соотношение $P+P \sim P$, выводим $P_2 \sim arg.\overline{res}.\mathbf{0} = P_1$. □

Рассмотрим теперь определение сильной эквивалентности для процессных выражений [9].

Определение 25 Пусть E и F – два процессных выражения, и $\mathrm{vars}\,(E) \cup \mathrm{vars}\,(F) \subseteq \{X_1, \ldots, X_n\}$. Мы будем считать выражения E и F *сильно эквивалентными* и писать $E \sim F$, если для любых процессов $P_1, \ldots, P_n \in \mathcal{P}$ выполнено соотношение

$$E\{P_1/X_1, \ldots, P_n/X_n\} \sim F\{P_1/X_1, \ldots, P_n/X_n\}$$

□

Следующий результат гарантирует, что при замене определяющего выражения константы на сильно эквивалентное ему сама константа до замены будет эквивалентна константе после замены. Это утверждение в данном пособии дается без доказательства.

Утверждение 21 ([Mil89], стр. 99) *Пусть E и F – два процессных выражения, X – процессная переменная, причем $\mathrm{vars}\,(E) \subseteq \{X\}$, $\mathrm{vars}\,(F) \subseteq \{X\}$. Пусть константы A и B определены как*

$$A \stackrel{def}{=} E\{A/X\} \qquad \textit{и} \qquad B \stackrel{def}{=} E\{B/X\}$$

Пусть также $E \sim F$. Тогда $A \sim B$.

Пример 6 Рассмотрим две процессные константы $Chan$ и $Chan'$:

$$\begin{array}{rcl} Chan & \stackrel{def}{=} & in.\overline{out}.Chan \\ Chan' & \stackrel{def}{=} & in.\big(\overline{out}.Chan' + \overline{out}.(Chan' + Chan')\big) \end{array}$$

Покажем, что $Chan \sim Chan'$. Пусть X – процессная переменная. Рассмотрим два процессных выражения E и E':

$$\begin{array}{rcl} E & = & in.\overline{out}.X \\ E' & = & in.\big(\overline{out}.X + \overline{out}.(X + X)\big) \end{array}$$

Пользуясь соотношением $P + P \sim P$ (пункт 1 утверждения 15), нетрудно доказать, что $E \sim E'$. По утверждению 21 отсюда следует $Chan \sim Chan'$. □

[9] В этом определении рассматриваются только процессные выражения, содержащие конечное число процессных переменных. Для выражений с бесконечным числом переменных мы эквивалентность определять не будем. В книге [Mil89] на странице 98 определяется эквивалентность для произвольных процессных выражений.

3.5 Наблюдаемая эквивалентность

В этом разделе будет определено отношение *наблюдаемой эквивалентности* состояний в размеченных системах переходов. Это отношение позволяет определенным образом игнорировать внутренние действия (τ) при сравнении состояний.

Для того, чтобы определить наблюдаемую эквивалентность, введем несколько дополнительных обозначений. Пусть задана некоторая размеченная система переходов $L = (S, R)$. Определим ряд бинарных отношений на множестве S. Пусть $\alpha \in Act$. Тогда

$$\begin{array}{ccl} \stackrel{\alpha}{\longrightarrow} & \stackrel{def}{=} & \{(s_1, s_2) \mid s_1 \stackrel{\alpha}{\longrightarrow} s_2\} \\ \stackrel{\varepsilon}{\Longrightarrow} & \stackrel{def}{=} & (\stackrel{\tau}{\longrightarrow})^* \\ \stackrel{\alpha}{\Longrightarrow} & \stackrel{def}{=} & (\stackrel{\varepsilon}{\Longrightarrow}) \circ (\stackrel{\alpha}{\longrightarrow}) \circ (\stackrel{\varepsilon}{\Longrightarrow}) \end{array}$$

Здесь, как и ранее, символ ε обозначает пустую последовательность действий, через $E_1 \circ E_2$ обозначено произведение отношений E_1 и E_2, а E^* есть рефлексивно-транзитивное замыкание отношения E.

Отношение $\stackrel{\alpha}{\longrightarrow}$ мы будем называть *отношением перехода с выполнением действия* α. Если $s_1, s_2 \in S$, то $s_1 \stackrel{\varepsilon}{\Longrightarrow} s_2$ равносильно существованию такого пути из s_1 в s_2, что при всех переходах этого пути выполняется действие τ. Формально это можно записать так:

$$\exists\, t_1, \ldots, t_n \in S \,: n \geq 1,\, s_1 = t_1,\, s_n = t_n, \forall\, i \,:\, (1 \leq i < n) \to (s_i \stackrel{\tau}{\longrightarrow} s_{i+1})$$

Если выполнено соотношение $s_1 \stackrel{\varepsilon}{\Longrightarrow} s_2$, то мы будем говорить, что s_2 является *ε-потомком* s_1.

Соотношение $s_1 \stackrel{\alpha}{\Longrightarrow} s_2$ выполнено тогда и только тогда, когда существуют такие $t_1, t_2 \in S$, что $s_1 \stackrel{\varepsilon}{\Longrightarrow} t_1$, $t_1 \stackrel{\alpha}{\longrightarrow} t_2$ и $t_2 \stackrel{\varepsilon}{\Longrightarrow} s_2$. В этом случае мы будем говорить, что что s_2 является *α-потомком* s_1.

Если $\alpha \in Act$, то через $\widehat{\alpha}$ мы будем обозначать

- пустую последовательность ε, если $\alpha = \tau$, и
- α в противном случае.

Определение 26 Пусть $L = (S, R)$ есть размеченная система переходов, и $E \subseteq S^2$. Отношение E называется *наблюдаемым* (или *слабым*) *бимоделированием*, если для любых состояний $(s_1, s_2) \in E$ выполнены следующие условия:

1. $\forall \alpha \in Act,\, s_1' \in S \,:\, s_1 \stackrel{\alpha}{\longrightarrow} s_1' \;\Rightarrow\; \exists s_2' \in S \,:\, s_2 \stackrel{\widehat{\alpha}}{\Longrightarrow} s_2' \,\wedge\, (s_1', s_2') \in E$
2. Второе условие симметрично первому:

$$\forall \alpha \in Act,\, s_2' \in S \,:\, s_2 \stackrel{\alpha}{\longrightarrow} s_2' \;\Rightarrow\; \exists s_1' \in S \,:\, s_1 \stackrel{\widehat{\alpha}}{\Longrightarrow} s_1' \,\wedge\, (s_1', s_2') \in E$$

Если $s_1, s_2 \in S$ – два состояния, то они называются *наблюдаемо* (или *слабо*) *эквивалентными*, если существует хотя бы одно наблюдаемое бимоделирование, содержащее пару (s_1, s_2). Наблюдаемая эквивалентность состояний s_1 и s_2 обозначается через $s_1 \approx s_2$. □

Отметим, что определение наблюдаемого бимоделирования отличается от определения сильного бимоделирования (определение 11) заменой отношения $\xrightarrow{\alpha}$ на $\xRightarrow{\widehat{\alpha}}$ справа от знака импликации в первом и втором условиях. Если P и Q – два процесса CCS, то мы будем считать их наблюдаемо эквивалентными и писать $P \approx Q$, если они наблюдаемо эквивалентны как состояния РСП $(\mathcal{P}, \mathcal{R})$ процессов CCS.

Пример 7 Пусть $P = a.\mathbf{0}$, и $Q = \tau.a.\tau.\mathbf{0}$. Покажем, что $P \approx Q$. Рассмотрим следующее отношение E:

$$E = \big\{(P,Q), (P, a.\tau.\mathbf{0}), (\mathbf{0}, \tau.\mathbf{0}), (\mathbf{0}, \mathbf{0})\big\}$$

Нетрудно убедиться, что E – наблюдаемое бимоделирование, откуда следует $P \approx Q$. □

Наблюдаемая эквивалентность обладает рядом свойств, аналогичных свойствам сильной эквивалентности. Следующие ее свойства доказаны в работе [Mil89] на странице 110:

Утверждение 22 *(Ср. с утверждением 2) Пусть $L = (S, R)$ – размеченная система переходов. Тогда отношение $\approx$ наблюдаемой эквивалентности состояний L обладает следующими свойствами:*

1. *$\approx$ является отношением эквивалентности на S.*
2. *$\approx$ является наблюдаемым бимоделированием на L.*
3. *$\approx$ является максимальным наблюдаемым бимоделированием на L, то есть $E \subseteq L$ для любого наблюдаемого бимоделирования E на L.*

Список литературы

[Bru97] G. Bruns. *Distributed Systems Analysis with CCS.* Prentice-Hall, 1997.

[Hoa89] C. Hoare. *Взаимодействующие последовательные процессы.* Мир, Москва, 1989.

[Mil80] R. Milner. *A Calculus of Communicating Systems*, volume 92 of *Lecture Notes in Computer Science.* Springer-Verlag, 1980.

[Mil89] R. Milner. *Communication and Concurrency.* Prentice-Hall, 1989.

[Mil90] Robin Milner. Operational and algebraic semantics of concurrent processes. In J. van Leewen, editor, *Handbook of Theoretical Computer Science*, volume B: Formal Models and Semantics, chapter 19, pages 1201–1242. The MIT Press, New York, NY, 1990.