РТУ МИРЭА

Теория сетей Петри и моделирование систем

Лекция 4

Способы описания сетей Петри

Существует несколько подходов к описанию СП:

- матричное описание;
- алгебраическое описание;
- описание на основе базовых фрагментов;
- графическое описание.

Способы описания сетей Петри

Алгебраическое описание сетей Петри

Алфавит языка:

буквы: N,T,Q;
специальные знаки: ";",":",",","$","+","*","-",">",".","#', "(", ")", "g", "h", "^";
цифры: 0,1,2,3,4,5,6,7,8,9;
пробел

Способы описания сетей Петри

Алгебраическое описание сетей Петри

Грамматика языка

< иерархическая сеть >:: = < идентификатор сети > : < описание сети>#
< описание сети > :: = < выражение > < список ИПР > | < выражение >
| < список ИПР >
< список ИПР > :: = < список ИПР > | < ИПР >
< ИПР > :: = < идентификатор ИПР> : < описание сети >
< выражение > :: = < терм > < операция > < выражение > | < операция >
< выражение > | < терм >
< терм > :: = (< выражение >) | < идентификатор сети > |
< идентификатор ИПР > | < идентификатор перехода >
< операция > :: = <число > > | , | ; | $ | + | * | - | ^ | <число>q | <число>h
< идентификатор сети > :: = N < число >
< идентификатор ИПР> :: = Q < число >
< идентификатор перехода > :: = T < указатель перехода >
< указатель перехода > :: = < число > | N < число> . < число > | Q
< число > . < число >
< число > :: = < цифра > < число > | < цифра >
< цифра > :: = 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Способы описания сетей Петри

Описание сетей Петри на основе базовых фрагментов

Определение 3. *Базовой вершиной-переходом* ***vt*** СП **N** , где *t* ∈ **T**, назовем фрагмент СП, включающий переход *t* и все позиции, для которых **I**(*p,t*) >=1 и **O***(p,t)*>=1 .

Определение 4. *Базовой вершиной-позицией* ***vp*** СП **N** , где *p* ∈ **P**, назовем фрагмент СП, включающий позицию *p* и все переходы, для которых **I(***p,t***)** >=1 и **O(***p,t***)** >=1 .

Вершину-переход (***vt***) и вершину-позицию (***vp***) в дальнейшем будем называть *базовыми фрагментами*.
Рассмотрим следующую теорему.

Теорема. СП **N = (P, T, I, O, μ0)** считается заданной, если заданы множества **P**, **T** и отображение **Г**, которое может быть определено:

- либо множествами входных элементов для каждой вершины **Г**=*{pre(**bi**)}*, где ***bi*** ∈ **P** ∪ **T** и *i* = $\overline{1,|P|+|T|}$;

- либо множествами выходных элементов для каждой вершины **Г**=*{post(**bi**)}*, где ***bi*** ∈ **P** ∪ **T** и *i* = $\overline{1,|P|+|T|}$.

Методы анализа сетей Петри

Задачи анализа

Одним из важнейших свойств СП, моделирующей реальное устройство, является *безопасность*. Известно, что если позиция безопасна, то число меток в ней равно 0 или 1. Безопасность - это частный случай более общего свойства *ограниченности*.

Другим важным свойством СП является наличие или отсутствие *тупиковых ситуаций*. Тупики в реальных системах возникают при распределении ресурсов между взаимодействующими процессами и служат предметом многих исследований в области сложных систем. В СП-модели аналогом тупиковых ситуаций являются тупиковые разметки.

Большинство задач, к которым мы до сих пор обращались, касается достижимости разметки.

Методы анализа сетей Петри

Задачи анализа

Задача достижимости формулируется следующим образом: для СП **N** с начальной разметкой $\mu0$ и разметкой μ' определить $\mu0 \to \mu'$.

Задача достижимости является основной задачей анализа СП. Многие другие задачи анализа можно сформулировать в терминах задачи достижимости.

В настоящее время наиболее широко используются два метода анализа СП, которые позволяют решить некоторые из перечисленных задач:

1 - построение дерева достижимых разметок,

2 - метод основанный на решении матричных уравнений.

Методы анализа сетей Петри

Анализ сетей Петри на основе дерева достижимости

Множество достижимых разметок **R(N)** в СП **N** можно представить в виде *дерева достижимости*. Данное дерево представляет собой ориентированный граф, множество вершин которого образовано множеством **R(N),** причем из вершины μ в вершину μ' ведет дуга, помеченная символом перехода *t*, если и только если

$$\mu \xrightarrow{t} \mu'.$$

В общем случае для любой СП с бесконечным множеством достижимых разметок соответствующее дерево также должно быть бесконечным.

Для превращения дерева в полезный инструмент анализа необходимо найти средства ограничения его до конечного размера. Для этого необходимо найти средства, которые ограничивают введение новых разметок, называемых граничными.

Методы анализа сетей Петри

Анализ сетей Петри на основе дерева достижимости

Будем использовать следующие приемы:

1) разметки, в которых нет разрешенных переходов (тупиковые разметки), будем называть *терминальными* вершинами;
2) выделим разметки, ранее встречавшиеся в дереве. Такие разметки будем называть *дублирующими* вершинами. Если встретилась дублирующая разметка, то никакие последующие маркировки рассматривать нет необходимости - все они будут порождены из места первого появления дублирующей разметки в дереве.
3) выделим разметки, в которых существуют позиции, увеличивающие число меток в результате срабатывания последовательности переходов ν. Следовательно, разметка данных позиций может быть неограниченно большой. Представим бесконечное число меток, формирующихся в позиции в результате циклов описанного типа, с помощью специального символа, который означает "бесконечность". Для любого постоянного α определим:

$$\omega + \alpha = \omega;\ \omega - \alpha = \omega;\ \alpha << \omega.$$

Дизайн И. Гайдель 2007

Методы анализа сетей Петри

Алгоритм построения дерева достижимости

Каждая вершина *i* дерева связывается с *расширенной разметкой* μi .

В расширенной разметке число меток в позиции может быть либо неотрицательным целым, либо ω .

Каждая вершина классифицируется или как *граничная*, *терминальная*, *дублирующая* вершина, или как *внутренняя*.

Граничными являются вершины, которые еще не обработаны алгоритмом. После обработки граничные вершины становятся либо терминальными, либо дублирующими, либо внутренними.

Методы анализа сетей Петри

Алгоритм построения дерева достижимости

Алгоритм начинает работу с определения начальной разметки (корневой вершины). До тех пор, пока имеются граничные вершины, они обрабатываются алгоритмом.

Пусть ***x*** - граничная вершина, которую необходимо обработать и с которой связана разметка $\mu(x)$:

1. Если в дереве имеется другая вершина ***y***, не являющаяся граничной, и с ней связана разметка $\mu(y) = \mu(x)$, то вершина ***x*** дублируется.

2. Если для разметки $\mu(x)$ ни один из переходов неразрешим, то есть $\mu(x)$ - тупиковая разметка, то ***x*** - терминальная вершина.

3. Для любого перехода ***ti*** из множества **T**, разрешенного в разметке $\mu(x)$, создать новую вершину *z* дерева достижимости. Разметка $\mu(z)$, связанная с этой вершиной, определяется для каждой позиции ***pi*** следующим образом:

а) если $\mu i(x) = \omega$, то $\mu i(z) = \omega$;

б) если на пути от корневой вершины к вершине ***x*** существует вершина ***y***, такая, что $\mu(y) \xrightarrow{t_i} \mu(x)$, $\mu(y) < \mu(x)$ и $\mu i(y) < \mu i(x)$, то $\mu i(z) = \omega$;

в) в противном случае $\mu i(z) = \mu i(x)$. Дуга, помеченная ***tj***, направлена от вершины ***x*** к вершине ***z***. Вершина ***x*** переопределяется как внутренняя, вершина ***z*** становится граничной.

4. Когда все вершины дерева становятся терминальными, дублирующими или внутренними, алгоритм останавливается.

Методы анализа сетей Петри

Матричные методы анализа

Показано, что если СП живая и ограниченная, то она должна быть последовательной и инвариантной. Данные свойства недостаточны для утверждения живости и ограниченности СП. Однако их полезно проверить исходя из матриц инцидентности, так как если одно из этих свойств не подтверждается, то можно заключить, что описываемая система содержит некоторые недоработки.

Методы анализа сетей Петри

Матричные методы анализа

Инвариантные и последовательные сети Петри . Введем в рассмотрение матрицу **C**, которая получается следующим образом:

$$\mathbf{C = O - I} \ .$$

Пусть размерность **C** равна $n \times m$, где m и n мощности множеств **P** и **T** .
Рассмотрим матричные уравнения :

$$\mathbf{C} * \boldsymbol{x} = \mathbf{0} \ ; \qquad (1)$$

$$\boldsymbol{y} * \mathbf{C} = \mathbf{0} \ , \qquad (2)$$

где ***x*** и ***y*** - векторы, размерность которых равна *n* и *m* соответственно.

Вектор ***y***, удовлетворяющий решению уравнения (1) и все элементы которого положительны, называется *p-цепью*; *p*-цепь, все элементы которой больше нуля, называется *полной p-цепью*.

Аналогично на основе уравнения (2) определяются понятия *t-цепи* и *полной t-цепи*.

СП, для которой существует полная *p*-цепь, называется *инвариантной*. СП, для которой существует полная *t*-цепь, называется *последовательной* .

Методы анализа сетей Петри

Матричные методы анализа.
Исследование сети Петри на живость и безопасность.

Описываемый метод анализа ИСП основан на определении свойств инвариантности и последовательности. Суть метода анализа ИСП заключается в нахождении для каждого ИПР СП решений уравнений (1) и (2) на множестве положительных целых чисел.

В общем случае множество решений данных уравнений может быть бесконечным. Так как целью решения уравнений является определение существования полных *p*- и *t*-цепей, то для сокращения времени вычислений воспользуемся следующим приемом. Пусть элементы множества $\mathbf{Z}=\{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_k\}$ являются решениями уравнения (1) (или (2)), причем элементы вектора $\boldsymbol{z}_i$ могут принимать значения только из набора {0,1}.

Методы анализа сетей Петри

Матричные методы анализа.
Исследование сети Петри на живость и безопасность.

Теорема. Вектор $\bar{z} = \mathbf{z_1} + \mathbf{z_2} + \ldots + \mathbf{z_k}$ является *t*-цепью (*p*-цепью).

Данная теорема позволяет значительно упростить процесс нахождения полных *p*- и *t*-цепей. Алгоритм поиска сводится к двум шагам:

1) определение множества $\mathbf{Z} = \{\mathbf{z_1}, \mathbf{z_2}, \ldots, \mathbf{z_k}\}$ для уравнения (1) (или (2));

2) получение вектора $\bar{z}$. Вид вектора $\bar{z}$ дает ответ на вопрос о существовании полных *p*- или *t*-цепей.

Спасибо за внимание!