

# РЕАЛИЗАЦИЯ АЛГОРИТМА ПРЕОБРАЗОВАНИЯ НЕОРДИНАРНОЙ СЕТИ ПЕТРИ В ОРДИНАРНУЮ

О.В. Усатюк, С.Л. Кривый

Институт кибернетики им. Глушкова НАН Украины,  
email: [oleg\\_usa@parus.com.ua](mailto:oleg_usa@parus.com.ua)

О.В. Усатюк, С.Л. Кривый Реалізація алгоритму перетворення неординарної мережі Петрі в ординарну. Описана програмна реалізація алгоритму перетворення неординарної мережі Петрі в ординарну. В основу програми покладено алгоритм, що був запропонований Хаком. Алгоритм реалізовано в програмному забезпеченні "Перетворювач мереж Петрі", створеному на мові програмування C++Builder.

О.В. Усатюк, С.Л. Кривый Реализация алгоритма преобразования неординарной сети Петри в ординарную. Описана программная реализация алгоритма преобразования неординарной сети Петри в ординарную. В основу программы положен алгоритм преобразования, предложенный Хаком. Алгоритм реализован в программном обеспечении «Преобразователь сетей Петри», созданному на языке программирования C++Builder.

O.V. Usatyuk, S.L. Krivoi Realisation of transformation algorithm of non-ordinary Petri Net into ordinary. An realization of transformation non-ordinary Petri net into ordinary. A basis of the program is the algorithm, offered by Hak. The given algorithm realized in the software «The Converter of networks of Petri», created on the programming language C ++ Builder.

## Введение

Построение модели дискретной системы – это важный этап в представлении абстрактных событий, которых может быть большое множество (например: выполнение операторов программы, переходы триггера из одного состояния в другое или прерывания операционной системы). Формализованное представление какой-нибудь дискретной системы является довольно сложным процессом, поскольку реальная система функционирует в реальном времени, т.е. события в рассматриваемой системе происходят в некоторые моменты времени и длится некоторое время. Поэтому ведение времени в дискретных системах обычно заменяется на причинно - следственные связи между событиями. Модели такого типа называются *асинхронными* [1], к ним также принадлежат и сети Петри. Одной из главных особенностей сетей Петри является то, что анализ некоторой построенной по описанной схеме системы позволяет выявить ее достоинства или недостатки, узкие места, которые в последствии можно устранить, что в свою очередь позволяет использовать сети Петри для моделирования и анализа в различных отраслях науки и техники. Особый интерес в этой области представляет некоторый подкласс сетей Петри – так называемые ординарные сети, о котором речь пойдет ниже.

## 1. Основные определения

Далее приводятся основные необходимые формальные определения.

**Определение 1.** Сетью будем называть тройку  $(P, T, F)$ , где

$P$  – непустое множество элементов сети, называемых местами,

$T$  – непустое множество элементов сети, называемых переходами,

$F \subseteq P \times T \cup T \times P$  – отношение инцидентности, и для  $(P, T, F)$  выполнены следующие условия: A1)

$P \cap T = \emptyset$  (множества мест и переходов не пересекаются);

A2)  $(F \neq \emptyset) \wedge (\forall x \in P \cup T, \exists y \in P \cup T : xFy \vee yFx)$  (т.е. любой элемент сети инцидентен хотя бы одному элементу другого типа);

A3) если для произвольного элемента сети  $x \in X$  обозначить через  $\cdot x$ , множество его входных элементов  $\{y \mid yFx\}$ , а через  $x \cdot$  – множество его выходных элементов  $\{y \mid xFy\}$ , то  $\forall p_1, p_2 \in P : (\cdot p_1 = \cdot p_2) \wedge (p_1 \cdot = p_2 \cdot) \Rightarrow (p_1 = p_2)$ , (т.е. сеть не содержит пары мест, которые инцидентны одному и тому же множеству переходов).

Графическим представлением сети служит в общем случае двудольный ориентированный граф с двумя типами вершин; вершины места изображаются кружочками, вершины-переходы – барьерами. Из вершины  $x$  в вершину  $y$  ведет дуга, тогда и только тогда, когда  $xFy$ .

**Определение 2.** Сетью Петри будем называть набор  $N=(P, T, F, W, M_0)$ , где  $(P, T, F)$  – конечная сеть (множество  $X = P \cup T$  конечно), а  $W : F \rightarrow N \setminus \{0\}$  и  $M_0 : P \rightarrow N$  – две функции, называемые соответственно кратностью дуг и начальной разметкой. Первая сопоставляет каждой дуге число  $p > 0$  (кратность дуги). Если  $p > 1$ , то в графическом представлении сети число  $p$  выписывается рядом с короткой чертой, пересекающей дугу. Часто такая дуга будет также заменяться пучком из  $p$  дуг, соединяющих соответствующие элементы сети. Условимся никак не отмечать кратность дуг, равную 1. Такую сеть будем называть

**ординарной.**[4] Вторая функция сопоставляет каждому месту  $p \in P$  некоторое число  $M_0(p) \in N$  (разметка места). В графическом представлении сети разметка места  $p$  изображается помещением в вершину-кружок числа  $M_0(p)$  или, если это число невелико, соответствующего числа точек (фишек).

Функционирование сети Петри описывается формально с помощью множества последовательностей срабатываний и множества достижимых в сети разметок. Эти понятия определяются через правила срабатывания переходов сети.

На основе отношения инцидентности  $F$  и функции кратности дуг  $W$  можно ввести функцию инцидентности  $F : P \times T \cup T \times P \rightarrow N$ , которая определяется как

$$F(x, y) = \begin{cases} n, & \text{если } xFy \wedge (W(x, y) = n), \\ 0, & \text{если } \neg(xFy). \end{cases}$$

Если места сети упорядочены, то можно каждому переходу  $t$  сопоставить два целочисленных вектора  $\cdot F(t)$  и  $F \cdot (t)$  длиной  $n$ , где  $n = |P|$ :

$$\cdot F(t) = (b_1, \dots, b_n), \quad \text{где } b_i = F(p_i, t),$$

$$F \cdot (t) = (b_1, \dots, b_n), \quad \text{где } b_i = F(t, p_i).$$

Переход  $t$  может сработать при некоторой разметке  $M$  сети  $N$ , если  $F'(t, p)$ , т.е. каждое входное место  $p$  перехода  $t$  имеет разметку, не меньшую, чем кратность дуги, соединяющей  $p$  и  $t$ .

Для **ординарной** сети Петри условие срабатывания перехода означает, что любое входное место этого перехода имеет ненулевую разметку, т.е. содержит хотя бы одну фишку.

Другими словами, если все дуги сети имеют одну и ту же кратность 1, то такая сеть относится к подклассу ординарных сетей. Наличие в сетях Петри дуг с кратностью, большей, чем единица, позволяет естественно моделировать реальные дискретные системы, но во многих случаях теоретического анализа сетей удобнее ограничиться рассмотрением ординарных сетей.

## 2. Алгоритм построения ординарной сети Петри

Вопрос о том, насколько результаты исследования ординарных сетей переносимы на общий случай сетей разрешил Хак [2]. Он показал, что подкласс ординарных сетей не является существенным сужением класса сетей Петри, и по отношению к большинству своих сетей оба класса оказываются эквивалентными в том смысле, что для сети Петри с заданным набором свойств можно построить ординарную сеть, обладающую тем же набором свойств [3]. Далее представлено предложенное Хаком преобразование произвольной сети Петри  $N=(P, T, F, W, M_0)$  в ординарную сеть  $N'=(P', T', F', M_0)$ , и состоит оно в следующем:

1) Для каждого места  $p \in P$  определяется максимальная кратность  $n(p)$  дуг, инцидентных этому месту по формуле  $n(p) = \max_{t \in T} (F(p, t) + F(t, p))$ .

2) Каждому месту  $p \in P$  будет соответствовать в сети  $N'$  множество  $P'(p)$  из  $n(p)$  мест  $p^1, p^2, \dots, p^{n(p)}$ , где  $n(p)$  – определенная выше максимальная кратность дуг для места  $p$ . Таким образом, общее число мест в  $P'$  равно сумме максимальных кратностей для всех мест из  $P$ , т.е.

$$P' = \bigcup_{p \in P} P'(p).$$

3) Каждому переходу  $t \in T$  соответствует в  $T'$  единственный переход, обозначаемый тем же символом  $t$ , но в сети  $N'$  появляется также множество  $T'(p) = \{r_1, r_2, \dots, r_{n(p)}\}$  новых переходов, которые связывают места  $p^1, p^2, \dots, p^{n(p)}$  из множества  $P'(p)$  в кольцевую сеть. При этом, если  $n(p)=1$ , то новые переходы не вводятся. Таким образом,  $T' = T \cup (\bigcup_{p \in P} T'(p))$ .

4) Для каждой дуги сети  $N$ , связывающей место  $p$  с некоторым переходом  $t$  и имеющей кратность  $W(p,t)$ , заводятся  $W(p,t)$  дуг, связывающих  $t$  с местами  $p^1, p^2, \dots, p^{n(p)}$ . При этом распределение дуг в сети  $N'$  по местам  $p^1, p^2, \dots, p^{n(p)}$  происходит произвольным образом, лишь бы не возникали ситуации, когда переход и место связаны более, чем одной дугой [4].

Для последнего пункта следует сделать замечание, касательно распределения дуг, поскольку предлагаемый «произвольный» порядок все-таки должен быть как-то оговорен и формализован. В программном обеспечении, которое будет рассмотрено ниже, при реализации этого отдельно взятого пункта, был использован подход, который состоит в следующем: поскольку переход и место могут быть связаны только одной дугой в ординарной сети, то матрицы инцидентности  $F'(t, p)$  и  $F'(p, t)$  будут содержать только единичные коэффициенты. Так как количество переходов ( $t$ ) полученной матрицы  $F'$  равно  $|T| + \sum_{p \in P} n(p)$ , а количество

мест (p) равно  $x \in X$ , что, собственно, и определяет размерность полученной матрицы инцидентности, то

предлагается леворекурсивное заполнение коэффициентами соответствующих результирующих матриц инцидентности единичными коэффициентами. Заметим, что количество единичных коэффициентов в строках и столбцах матриц  $F'(t, p)$  и  $F'(p, t)$  для первых  $|T|$  переходов соответственно будет равно сумме коэффициентов строк и столбцов исходных матриц  $F(t, p)$  и  $F(p, t)$  для каждого из переходов. Также следует отметить тот факт, что оставшаяся часть результирующей матрицы инцидентности размерностью  $\sum_{p \in P} n(p) \times \sum_{p \in P} n(p)$  представляет собой диагональную матрицу с единичными коэффициентами. Причем если количество переходов исходной сети Петри было равно  $t$ , а количество мест равно  $p$ , и при этом количество переходов результирующей матрицы равно  $t' = t + p'$ , а мест  $p' = \sum_{p \in P} n(p)$ , то диагональные

матрицы будут сформированы следующим образом (для  $F'(t, p)$  и  $F'(p, t)$  соответственно):

для $F'(t, p)$					для $F'(p, t)$				
$\vdots$	$p'_1$	$p'_2$	$\dots$	$p'$	$\vdots$	$t-p'+1$	$t-p'+2$	$\dots$	$t$
$t-p'+1$	0	1	0	0	$p'_1$	1	0	0	0
$t-p'+2$	0	0	1	0	$p'_2$	0	1	0	0
$\vdots$	0	0	0	1	$\vdots$	0	0	1	0
$t$	1	0	0	0	$p'$	0	0	0	1

Как видно из представленных частей результирующих матриц инцидентности, в первой из них представлено закрытие кольцевой сети, что соответствует пункту 3 алгоритма построения ординарной сети Петри. Рассмотренный выше алгоритм нашел свое применение при создании программного обеспечения по преобразованию произвольной заданной сети Петри в соответствующую ей ординарную сеть, - о нем, а также об особенностях его использования пойдет речь в следующем разделе.

### 3. О реализации алгоритма преобразования

Описанный выше алгоритм преобразования для сетей Петри был реализован в программном обеспечении «Преобразователь сетей Петри», написанном на языке программирования C++Builder. Программа представляет собой Windows – приложение, позволяющее производить ввод сети Петри и производить расчет коэффициентов для ординарной сети Петри. Общий вид приложения показан на рис 1.

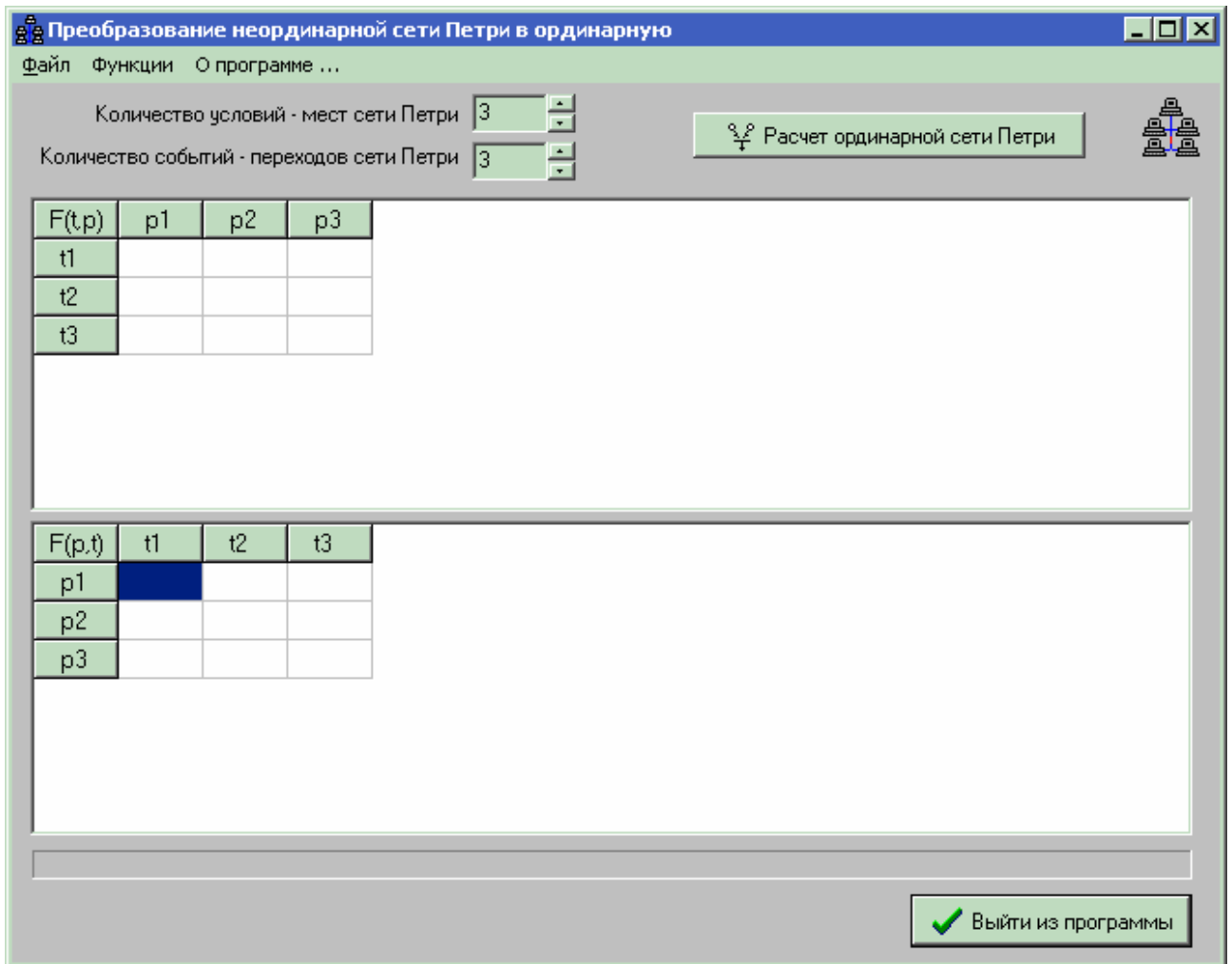


Рис. 1.

Как видно из рисунка, ввод коэффициентов матрицы инцидентности для произвольной сети Петри происходит в двух таблицах, размерность которых задается при помощи полей: «Количество условий-мест сети Петри», а также «Количество событий-переходов сети Петри». Увеличение или уменьшение значений в этих полях осуществляется посредством использования стрелок «вверх/вниз», а также можно вводить нужные значения с клавиатуры.

Перемещение по полям таблицы производится при помощи клавиш «влево/вправо» и «вверх/вниз», «Enter» - исправление значения. Есть возможность одновременной очистки всех введенных коэффициентов, для этого можно воспользоваться соответствующим пунктом главного меню «Функции», как показано на рис. 2.

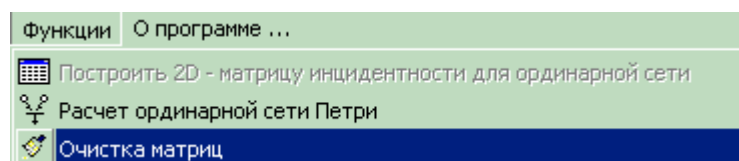


Рис. 2.

Программа также позволяет сохранять введенные коэффициенты матриц инцидентности в файл или загружать предварительно сохраненные сети Петри из файла. Осуществляется эта возможность при помощи пунктов основного меню «Файл» - рис. 3.

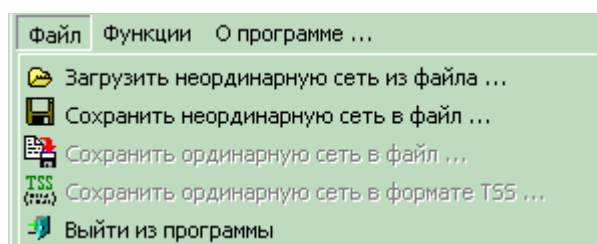


Рис. 3.

Неактивными пунктами меню на рисунке 3 можно будет пользоваться после выполнения работы алгоритма по преобразованию произвольной сети Петри в ординарную. Описание работы с этими пунктами меню будет дано ниже по тексту статьи.

На рис. 2 также есть подменю «Расчет ординарной сети Петри». Посредством выбора этого пункта меню или аналогичной кнопки на главной форме производится запуск расчета сети Петри, в процессе которого программа может выдавать некоторые информационные сообщения.

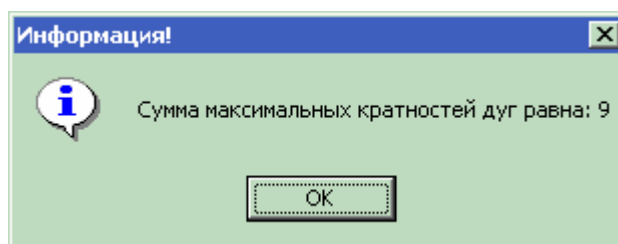


Рис. 4.

Сообщение на рис. 4 выдается для проведения возможного контроля за ходом работы алгоритма, поскольку информация о сумме максимальных кратностей дуг соответствует второму шагу работы алгоритма, изложенного выше.

Об успешном проведении расчета ординарной сети программа выдаст соответствующее сообщение типа «Расчет ординарной сети Петри произведен за 20 мсек». Таким образом пользователь может также получить информацию о затратах времени на проведение расчета ординарной сети. Это может быть полезно, когда необходимо проследить временную зависимость работы алгоритма при различных входных параметрах, т.е. при задании различных матриц инцидентности, соответствующих произвольным сетям Петри.



Рис. 5.

Следует отметить, что визуальное отображение полученных матриц может занимать довольно продолжительные интервалы времени, поэтому в программе предусмотрено соответствующее сообщение пока описанный процесс выполняется, – показано на рисунке 5.

После того как процесс заполнения таблиц с результирующими матрицами инцидентности уже завершился, пользователь видит форму, показанную на рис. 6. На форме также содержится информация о времени, затраченном на построение ординарной сети Петри. При желании можно также увидеть размерность матриц инцидентности, информация об этом содержится в левом верхнем углу каждой из таблиц, находящихся на форме. Этим работа с формой не заканчивается. Дело в том, что было бы весьма полезно получить матрицу инцидентности не в виде двух составляющих, а как единую целую, - в программе предусмотрена и такая возможность, хотя о ней речь пойдет ниже. Почему нельзя было сразу отобразить двухмерную матрицу инцидентности ординарной сети Петри, - потому, что при наличии в полученной сети полувидов они в такой матрице исчезают. Для этого и были показаны составляющие матрицы инцидентности на форме с результатами вычислений. На рассматриваемой форме есть возможность сохранения результатов вычислений, для чего необходимо воспользоваться пунктом контекстного меню «Сохранить ординарную сеть в файл...» (рис. 8), после этого появляется стандартное окно Windows, в котором указывается имя файла и путь, по которому нужно провести сохранение. После успешного сохранения программа выдаст соответствующее сообщение.

Есть также возможность получения двухмерной матрицы инцидентности для дальнейших исследований. Для вызова этой функции достаточно воспользоваться соответствующим пунктом контекстного меню «Получить 2D – матрицу инцидентности для ординарной сети Петри», как показано на рис. 7. После вызова этого пункта меню, пользователь получает новую форму (рис. 8) с соответствующей двухмерной матрицей инцидентности, причем также существует возможность сохранения полученных результатов в файл при помощи соответствующего пункта контекстного меню, причем сохранение производится в формате TSS (усеченное множество решений) [5].

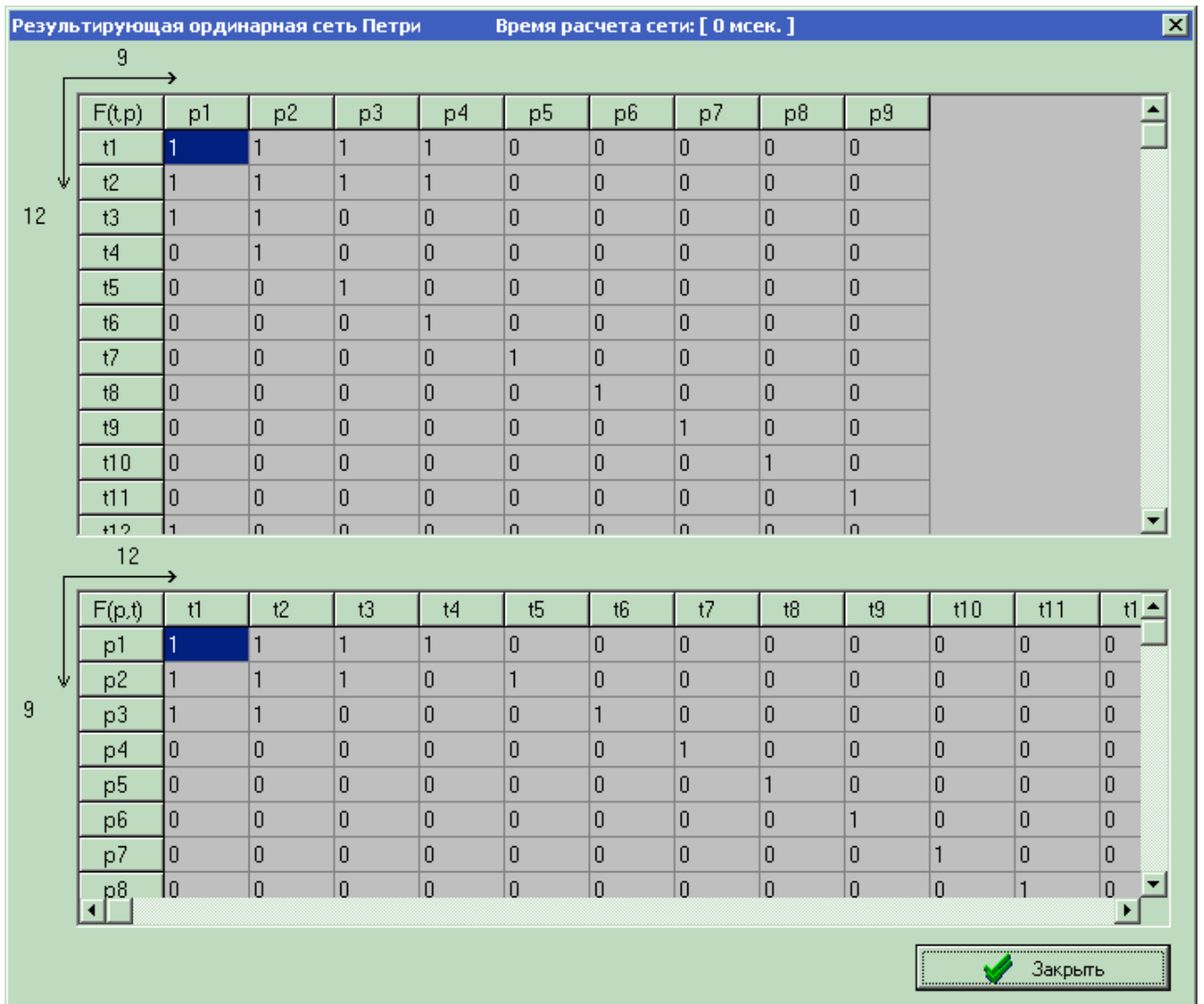


Рис. 6.

Пункт контекстного меню «Закреть» присутствует на всех выше и ниже указанных формах и работает одинаково - закрывает активную форму, аналогичного результата можно достичь путем использования соответствующей кнопки в правом нижнем углу формы.

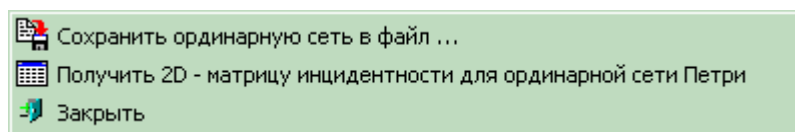


Рис. 7.

Двухмерная матрица инцидентности для сети Петри

F	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10	t11	t12
p1	0	0	0	-1	0	0	0	0	0	0	0	1
p2	0	0	0	1	-1	0	0	0	0	0	0	0
p3	0	0	0	0	1	-1	0	0	0	0	0	0
p4	1	1	0	0	0	1	-1	0	0	0	0	0
p5	0	0	0	0	0	0	1	-1	0	0	0	0
p6	0	0	0	0	0	0	0	1	-1	0	0	0
p7	0	0	0	0	0	0	0	0	1	-1	0	0
p8	0	0	0	0	0	0	0	0	0	1	-1	0
p9	0	0	0	0	0	0	0	0	0	0	1	-1

Закрывать

Рис. 8.

Возможность сохранения результатов вычислений, представленных на форме (рис. 8), осуществляется посредством вызова контекстного меню (рис. 9), а затем выбора пункта меню «Сохранить ординарную сеть в формате TSS». Эта возможность весьма полезна при дальнейшем анализе ординарной сети, например: при поиске в полученной матрице инцидентности Т-инвариантов, а также при произведении некоторых преобразований с матрицей для нахождения S-инвариантов.

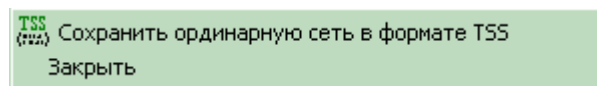


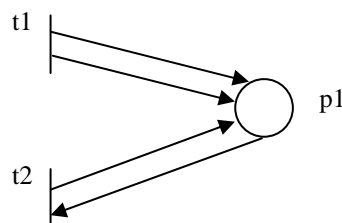
Рис. 9.

Представленное программное обеспечение содержит также большое множество проверок на введенные (или не введенные) коэффициенты или загружаемую из файла информацию, которые, в принципе, являются нормой для программ такого рода. Поэтому при отсутствии всех необходимых параметров процесс получения ординарной сети Петри может не начаться, - в этом случае следует четко следовать замечаниям программы. Наличие такого рода нюансов в программе изменяется от версии к версии, информацию о которой можно получить, воспользовавшись пунктом главного меню «О программе ...».

#### 4. Примеры работы программы.

Далее представлены некоторые примеры работы рассмотренного программного обеспечения с временными затратами на преобразование сетей Петри.

1).



Для заданной сети Петри получим такие первоначальные матрицы инцидентности:

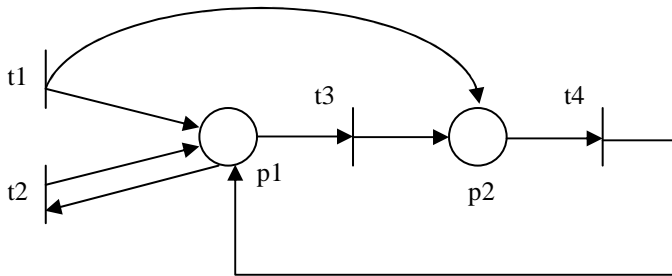
$$F(t, p) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \quad F(p, t) = \begin{pmatrix} 0 & 1 \end{pmatrix}.$$

Сумма максимальных кратностей дуг равна 2:  $n(p) = \max_p (2+0, 1+1) = \max_p (2,2) = 2$ .

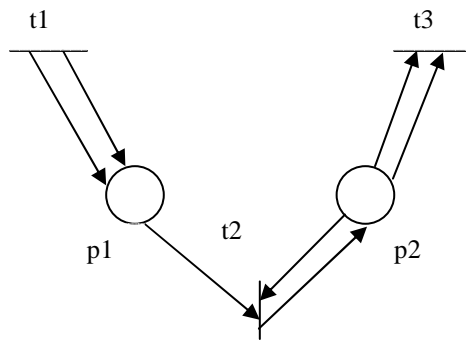
Результирующее количество мест  $p' = 2$  и переходов  $t' = n(p) + 2 = 4$ , при этом матрицы имеют вид:

$$F'(t, p) = \begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \text{ и } F'(p, t) = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Расчет ординарной сети в этом случае произведен за 0 мсек. Результирующая ординарная сеть Петри имеет вид:



2).



Для заданной сети Петри получим такие первоначальные матрицы инцидентности:

$$F(t, p) = \begin{pmatrix} 2 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}, F(p, t) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 2 \end{pmatrix}.$$

Сумма максимальных кратностей дуг равна 4:

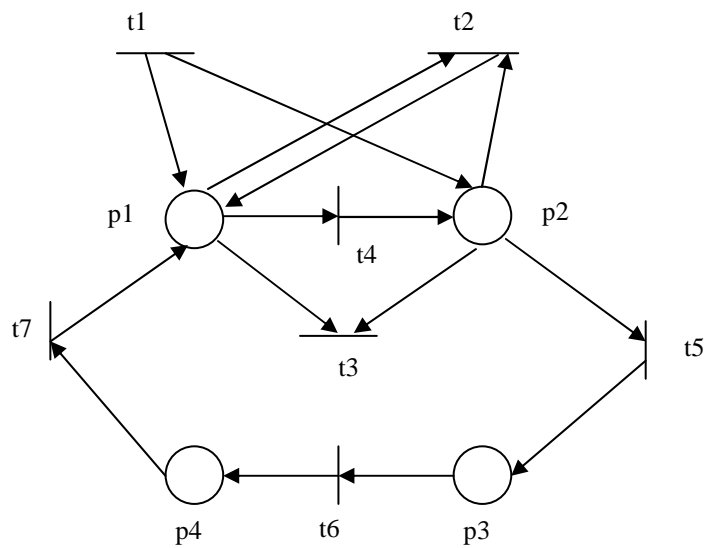
$$n(p) = \max_p \begin{pmatrix} 2+0 & 0+0 \\ 0+1 & 1+1 \\ 0+2 & 0+0 \end{pmatrix} = \max_p \begin{pmatrix} 2 & 0 \\ 1 & 2 \\ 2 & 0 \end{pmatrix} = (2,2) = 2+2 = 4.$$

Результирующее количество мест  $p' = 4$  и переходов  $t' = n(p) + 3 = 7$ , при этом матрицы имеют вид:

$$F'(t, p) = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix} \text{ и } F'(p, t) = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$



Расчет ординарной сети в этом случае произведен за 0 мсек. Результирующая ординарная сеть Петри имеет вид:



### Литература:

1. Genrich H., Lauterbach K., Thiagarajan P.S. Elements of general net theory. – In: Lecture Notes of Computer Science. Berlin: Springer-Verlag, 1979, p. 21-164.
2. Hack M. Decision problems for Petri nets and vector addition systems. – Project MAC Memo 59. Cambridge, 1975.
3. Hack M. Decidability questions for Petri nets. – Laboratory for Computer Science TR 161. Cambridge, 1976.
4. Котов В.Е. Сети Петри. М.: Наука. Главная редакция физико-математической литературы, 1984, с.11-58.
5. Усатюк О.В. Реализация инкрементной версии алгоритма построения минимального множества решений системы линейных уравнений на множестве натуральных чисел. Киев. УСиМ. – 2002. - №2, с.26-34.