Модел. и анализ информ. систем. Т. **21**, № **6** (2014) 18–30

УДК 519.71+004.021

Управляемые тупики в параллельных ресурсно-ограниченных потоках работ

Башкин В.А.,[1] Панфилова Н.Ю.

Ярославский государственный университет им. П. Г. Демидова
150000 Россия, г. Ярославль, ул. Советская, 14

e-mail: v_ bashkin@mail.ru ; lillian007@mail.ru

получена 15 сентября 2014

Ключевые слова: потоки работ, ресурс, бездефектность, параллельная композиция, тупик, верификация

Работа посвящена проблеме проверки правильной организованности (бездефектности) сетей потоков работ с ресурсами. Поток работ называется бездефектным, если он может быть корректно завершен от любого достижимого состояния. Рассматривается класс схем ресурсно-ограниченных потоков работ (RCWF-сетей), в которых экземпляры процесса могут использовать внешние ресурсы, но не могут за время своей жизни изменить их количество.

Две бездефектные RCWF-сети, использующие один и тот же набор ресурсов, могут быть запущены параллельно. Подобная параллельная композиция в некоторых случаях может порождать дополнительные тупики, вызванные взаимными блокировками. Мы исследуем проблему обнаружения потенциальных блокировок и предлагаем способы организации такого управления сетью, которое позволило бы их избегать.

1. Введение

Системы управления потоками работ (Workflow Management Systems) широко применяются для автоматизированной поддержки управления технологическими и бизнес-процессами, поскольку позволяют существенно снизить стоимость и время выполнения задач, а также повысить надежность и качество обслуживания. При помощи схем потоков работ можно описывать взаимодействие людей, ресурсов, устройств и потоков информации. В последние годы одним из стандартных способов формального представления workflow стали сети потоков работ [4, 1], представляющие собой особый класс сетей Петри.

Сеть потока работ (WF-сеть) представляет собой математическую абстракцию, которая может быть использована для верификации важнейшего поведенческого свойства моделируемой системы — её бездефектности (правильной организованности). Данное свойство гарантирует отсутствие тупиков, динамических тупиков, а

[1]Работа поддержана РФФИ (проект 12-01-00281).

также прочих аномалий поведения, которые могут быть выявлены без знания специфики предметной области (то есть только по логической схеме процесса). В настоящее время существует несколько понятий бездефектности (см. обзор [5]). Согласно классическому определению, бездефектная система должна быть способна корректно завершить свою работу после попадания в любое из достижимых состояний.

Схема потока работ состоит из набора элементарных задач, связанных между собой причинно-следственными (или хронологическими) связями. В совокупности такая схема может описывать различные процессы, происходящие внутри организации (управление, документооборот и т.п.). В реальном мире для выполнения конкретных задач могут требоваться дополнительные ресурсы — машины, рабочие руки и т.п. Концепция абстрактных ресурсов позволяет обнаруживать важные эквивалентности в пространстве состояний распределенных систем [8] и моделировать различные аспекты их поведения [2, 3].

Для адекватного отражения в моделях ресурсной составляющей исследователями были предложены различные ресурсные расширения базового формализма WF-сетей, порождающие различные расширенные определения бездефектности. В работах [6, 7] был описан особый класс WFR-сетей с разрешимой бездефектностью. В [13, 15] авторы определили более общий класс ресурсно-ограниченных сетей, названный ими RCWF-сетями (Resource-Constrained Workflow Nets). В таких моделях вводятся два естественных ограничения на ресурсы. Во-первых, все ресурсы, изначально доступные экземпляру процесса, должны быть по завершении работы возвращены им системе. Во-вторых, при любом достижимом состоянии текущее количество ресурсов не должно превышать начального. В работах [9, 10] был рассмотрен ещё более общий класс произвольных трансформаций ресурсов.

В работе [13] было доказано, что для RCWF-сетей с одним типом ресурсов свойство обобщенной бездефектности может быть эффективно проверено за полиномиальное время. В [15] была доказана разрешимость бездефектности для RCWF-сетей с произвольным количеством ресурсных позиций. Заметим, что при этом эффективного алгоритма не было получено, так как задача решалась сведением к проблеме домашнего состояния. Также осталась открытой проблема вычисления наименьшего объема ресурсов, при котором данная сеть бездефектна.

В данной статье мы рассматриваем композициональный подход к проблеме бездефектности. Изучаются возможности разбиения глобального процесса на независимые по управлению и зависимые по ресурсам параллельные подпроцессы (параллельные ветви потока работ). Мы используем естественное понятие параллельной композиции двух RCWF-сетей, имеющих общее множество ресурсных позиций. Параллелизм при этом может порождать новые тупиковые состояния, вызванные взаимными блокировками подпроцессов. Показано, что определенное увеличение начального ресурса позволяет избавиться от всех подобных тупиков (и прочих нарушений бездефектности). Таким образом, представлен основанный на декомпозиции метод вычисления нетривиального подмножества множества минимальных бездефектных ресурсов RCWF-сети.

Главным результатом статьи является метод управления блокировками для параллельных потоков работ. Показано, что при определённых условиях составная сеть может быть реструктурирована таким образом, чтобы в результате требовать не суммы, а объединения минимальных бездефектных ресурсов своих параллель-

ных подсетей. На практике это может позволить экономить значительную часть начальных ресурсов при сохранении свойства бездефектности.

Статья организована следующим образом. В Главе 2 приводятся основные определения и обозначения, касающиеся мультимножеств и сетей Петри. В Главе 3 формально определяются ресурсно-ограниченные сети Петри и их свойства бездефектности. Глава 4 посвящена исследованию достижимости в бездефектных RCWF-сетях. В Главе 5 вводится понятие параллельной композиции RCWF-сетей. Формулируются результаты, описывающие получение минимальных ресурсов составной сети из минимальных ресурсов её параллельных подсетей. В Главе 6 представлены методы управления блокировками, обеспечивающие бездефектность поведения составной сети. Первый из них может быть применен к произвольной паре бездефектных потоков работ, однако требует специального контроля за процессом во время его исполнения. Второй использует для контроля и избежания блокировок структуру самой сети, однако применим только к так называемым безопасным сетям. В Главе 7 приводятся некоторые выводы.

2. Предварительные сведения

Пусть S — конечное множество. *Мультимножеством* M над множеством S называется отображение $M : S \to Nat$, где Nat — множество неотрицательных целых чисел. Обозначим через $\mathcal{M}(S)$ множество всех конечных мультимножеств над S.

Операции и отношения теории множеств естественно расширяются на конечные мультимножества. Пусть $M_1, M_2, M_3 \in \mathcal{M}(S)$. Полагаем: $M_1 \subseteq M_2 \Leftrightarrow \forall s \in S : M_1(s) \leq M_2(s)$; $M_1 = M_2 + M_3 \Leftrightarrow \forall s \in S : M_1(s) = M_1(s) + M_2(s)$; $M_1 = M_2 \cup M_3 \Leftrightarrow \forall s \in S : M_1(s) = max\{M_1(s), M_2(s)\}$.

Сетью Петри называется набор $N = (P, T, F)$, где P — конечное множество позиций; T — конечное множество переходов, $P \cap T = \varnothing$; $F : (P \times T) \cup (T \times P) \to Nat$ — функция инцидентности (мультимножество дуг).

Разметкой (состоянием) сети N называется функция вида $M : P \to Nat$, сопоставляющая каждой позиции сети некоторое натуральное число (или ноль). Разметка может рассматриваться как мультимножество над множеством позиций сети, то есть элемент множества $\mathcal{M}(P)$. *Размеченной сетью Петри* называется пара (N, M_0), где $N = (P, T, F)$ — сеть Петри, $M_0 \in \mathcal{M}(P)$ — начальная разметка (количество ресурса в наличии при запуске сети).

Графически сеть Петри изображается как двудольный ориентированный граф. Вершины-позиции изображаются кружками и характеризуют локальные состояния сети, вершины-переходы изображаются прямоугольниками и соответствуют действиям. Дуги соответствуют элементам F. Позиции могут содержать маркеры (фишки), изображаемые черными точками. При разметке M в каждую позицию p помещается $M(p)$ фишек. Для перехода $t \in T$ через ${}^\bullet t$ и $t^\bullet$ обозначим мультимножества его входных и выходных позиций: $\forall p \in P\ {}^\bullet t(p) =_{def} F(p, t),\ t^\bullet(p) =_{def} F(t, p)$.

Переход $t \in T$ *активен (готов к срабатыванию)* при разметке M, если ${}^\bullet t \subseteq M$ (все входные позиции содержат достаточное число фишек). Готовый к срабатыванию переход t может *сработать*, порождая новую разметку $M' =_{def} M - {}^\bullet t + t^\bullet$

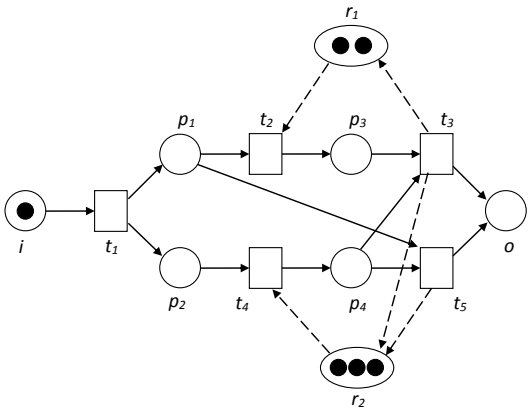


Рис. 1. RCWF-сеть

(обозн. $M \xrightarrow{t} M'$). Множество всех разметок, достижимых из начальной разметки M за ноль и большее число срабатываний, обозначается как $\mathcal{R}(N, M)$.

Сеть (N, M_0) называется *ограниченной,* если множество $\mathcal{R}(N, M_0)$ конечно. Сеть (N, M_0) называется *безопасной,* если $\forall M \in \mathcal{R}(N, M_0), p \in P$ выполняется $M(p) \leq 1$.

3. WF-сети с ресурсами

Ресурсно-ограниченной сетью потока работ (RCWF-сетью) называется набор $N = (P_c, P_r, T, F_c, F_r, i, o)$, где P_c — конечное множество управляющих позиций; P_r — конечное множество ресурсных позиций, $P_c \cap P_r = \emptyset$; T — конечное множество переходов, $P_c \cap T = P_r \cap T = \emptyset$; $F_c \ : \ (P_c \times T) \cup (T \times P_c) \to Nat$ — мультимножество управляющих дуг; $F_r \ : \ (P_r \times T) \cup (T \times P_r) \to Nat$ — мультимножество ресурсных дуг; $\forall t \in T \ \exists p \in P_c \ : \ F_c(p,t) + F_c(t,p) > 0$ (каждый переход инцидентен некоторой управляющей позиции); $i \in P_c$ и $o \in P_c$ — выделенные начальная и финальная позиции соответственно, где ${}^\bullet i = o^\bullet = \emptyset$; каждый элемент множества $P_c \cup T$ лежит на пути из i в o, состоящем из элементов $P_c \cup T$.

В отличие от обыкновенных сетей Петри, в RCWF-сетях позиции делятся на управляющие и ресурсные. Каждый переход связан хотя бы с одной управляющей позицией (что гарантирует отсутствие “неуправляемых” модификаций ресурсов). Заметим, что из последнего требования следует, что каждый переход обладает как минимум одной входной дугой от управляющей позиции и как минимум одной выходной дугой к управляющей позиции.

Разметка также делится на управляющую и ресурсную части. Мультимножество вида $c + r$, где $c \in \mathcal{M}(P_c)$ и $r \in \mathcal{M}(P_r)$, мы будем обозначать как $c|r$.

Для сети N *ресурсами* называются мультимножества над P_r.

На Рис. 1 приведён пример RCWF-сети. Ресурсные позиции r_1 и r_2 изображены эллипсами, ресурсные дуги — пунктирными стрелками.

Каждая RCWF-сеть $N = (P_c, P_r, T, F_c, F_r, i, o)$ содержит *управляющую подсеть* $N_c = (P_c, T, F_c, i, o)$, которая является RCWF-сетью с пустым множеством ресурсов.

Размеченной сетью $(N, i|r)$ называется сеть N вместе с некоторой начальной

разметкой $i|r$ (здесь i обозначает мультимножество, содержащее единственную фишку в начальной позиции i).

RCWF-сеть N называется (r)-*бездефектной* для некоторого ресурса $r \in \mathcal{M}(P_r)$, если для любой разметки $c|r'$ из множества $\mathcal{R}(N, i|r)$ выполняется неравенство $r' \leq r$, и, кроме того, $o|r \in \mathcal{R}(N, c|r')$. RCWF-сеть N *бездефектна,* если существует ресурс $r \in \mathcal{M}(P_r)$ такой, что N является (r')-бездефектной при любом $r' \geq r$.

Например, сеть на Рис. 1 является бездефектной, $(r_1 + r_2)$-бездефектной, и не является (r_1)-бездефектной.

Таким образом, бездефектность RCWF-сети означает, что, во-первых, данный поток работ может завершиться правильно при старте от любой достижимой разметки, и, во-вторых, добавление дополнительных начальных ресурсов не может нарушить этого свойства. В работе [15] было доказано, что бездефектность разрешима даже в более общем случае нескольких фишек в начальной позиции i.

Определение 1. *Для бездефектной RCWF-сети N через* $\mathbf{res}(N)$ *и* $\mathbf{mres}(N)$ *обозначим множества бездефектных и минимальных бездефектных ресурсов:*

- $\mathbf{res}(N) =_{def} \{r \in \mathcal{M}(P_r) \mid N \ \ (r + r') -$ *бездефектна для* $\forall r' \in \mathcal{M}(P_r)\}$;
- $\mathbf{mres}(N) =_{def} \{r \in \mathbf{res}(N) \mid \ \nexists r' \in \mathbf{res}(N) : r' < r\}$.

Очевидно, что множество $\mathbf{mres}(N)$ конечно. Например, для сети на Рис. 1 выполняется $\mathbf{mres}(N) = \{r_1 + r_2\}$.

Как было отмечено в [15], проблема построения $\mathbf{mres}(N)$ всё ещё открыта. В данной статье мы исследуем специфический подход к данной проблеме, основанный на параллельной композиции/декомпозиции RCWF-сетей.

4. Свойства бездефектных ресурсов

Следующее утверждение формализует известное свойство "корректного завершения" бездефектных потоков работ:

Факт 1. *Для любой (r)-бездефектной сети если $c|r \in \mathcal{R}(N, i|r)$, то $c = o \vee c \cap o = \emptyset$.*

Доказательство. Предположим противное: $o + m|r \in \mathcal{R}(N, i|r)$ для некоторого непустого m. Из второго требования для бездефектности имеем $o|r \in \mathcal{R}(N, o+m|r)$. Поскольку позиция o не имеет выходных дуг, получим $\emptyset|r \in \mathcal{R}(N, m|r)$. Но при этом каждый переход в N имеет хотя бы одну *выходную* управляющую позицию, следовательно, $m = \emptyset$ — противоречие. □

Другим достаточно очевидным фактом является бездефектность и ограниченность управляющей подсети:

Факт 2. *Для любой бездефектной сети $N = (P_c, P_r, T, F_c, F_r, i, o)$ и её управляющей подсети $N_c = (P_c, T, F_c, i, o)$ имеем: (1) N_c — $(\emptyset)$-бездефектна; (2) $(N_c, i|\emptyset)$ — ограничена; (3) если $c|\emptyset, c + c'|\emptyset \in \mathcal{R}(N, i|r)$, то $c' = \emptyset$.*

Доказательство. (1) Предположим противное: N_c не $(\emptyset)$-бездефектна. Поскольку N_c не содержит ресурсных позиций, для неё нарушена только вторая часть определения бездефектности: существует такая разметка $c|\emptyset$ из множества $\mathcal{R}(N_c, i|\emptyset)$, для которой выполняется $o|\emptyset \notin \mathcal{R}(N_c, c|\emptyset)$. Обозначим соответствующую последовательность переходов σ (то есть $i|\emptyset \xrightarrow{\sigma} c|\emptyset$).

Теперь рассмотрим N. Очевидно, найдется начальный ресурс r настолько большой, что $c|r' \in \mathcal{R}(N, i|r)$ для некоторого r' — достаточно сложить ресурсы, потребляемые всеми переходами последовательности σ.

С другой стороны, никакого ресурса x не должно быть достаточно для достижения финального состояния $o|y$ от $c|x$ (для любого y), поскольку финальное состояние недостижимо даже в не ограниченной ресурсами управляющей подсети N_c. Следовательно, N не бездефектна.

(2) В противном случае в $(N_c, i|\emptyset)$ было бы возможно бесконечное исполнение, содержащее бесконечное число различных разметок и, следовательно, пару разметок $c_1 < c_2$ таких, что $i \to c_1 \to c_2 \to \ldots$. Из бездефектности N_c следует $c_1 \xrightarrow{\sigma} o$ для некоторой последовательности переходов $\sigma \in T^*$. Но, поскольку $c_1 < c_2$, та же последовательность переходов возможна и при c_2 : $c_2 \xrightarrow{\sigma} o + (c_2 - c_1)$ — что противоречит свойству корректного завершения.

(3) Предположим противное. По свойству бездефектности существуют две последовательности переходов: $i|\emptyset \to c|\emptyset \to o|\emptyset$ и $i|\emptyset \to c + c'|\emptyset \to o|\emptyset$.

Из первой последовательности и свойства монотонности сетей Петри получим $i + c'|\emptyset \to c + c'|\emptyset \to o + c'|\emptyset$. Объединяя это со второй последовательностью, получим $i|\emptyset \to c + c'|\emptyset \to o + c'|\emptyset$ — что противоречит бездефектности. □

Каждому достижимому управляющему состоянию $c \in \mathcal{M}(P_r)$ соответствует единственное достижимое значение ресурса:

Лемма 1. *Если сеть N бездефектна, $r \in \mathbf{res}(N)$ и $c|r_1, c|r_2 \in \mathcal{R}(N, i|r)$, то $r_1 = r_2$.*

Доказательство. Предположим противное: пусть $r_1 \neq r_2$.

Рассмотрим некоторое $r' = r_1 + \delta_1 = r_2 + \delta_2$. Из $r_1 \neq r_2$ следует $\delta_1 \neq \emptyset$ или $\delta_2 \neq \emptyset$. Кроме того, $\delta_1 \neq \delta_2$.

Имеем $i|r \to c|r_1 \to o|r$ и, следовательно, по свойству монотонности сетей Петри $i|r + \delta_1 \to c|r_1 + \delta_1 \to o|r + \delta_1$. Аналогично, $i|r + \delta_2 \to c|r_2 + \delta_2 \to o|r + \delta_2$. Но $r_1 + \delta_1 = r' = r_2 + \delta_2$ и, следовательно, имеем $i|r + \delta_1 \to c|r' \to o|r + \delta_2$. Из $(r + \delta_1)$-бездефектности должно следовать $\delta_1 = \delta_2$ — противоречие. □

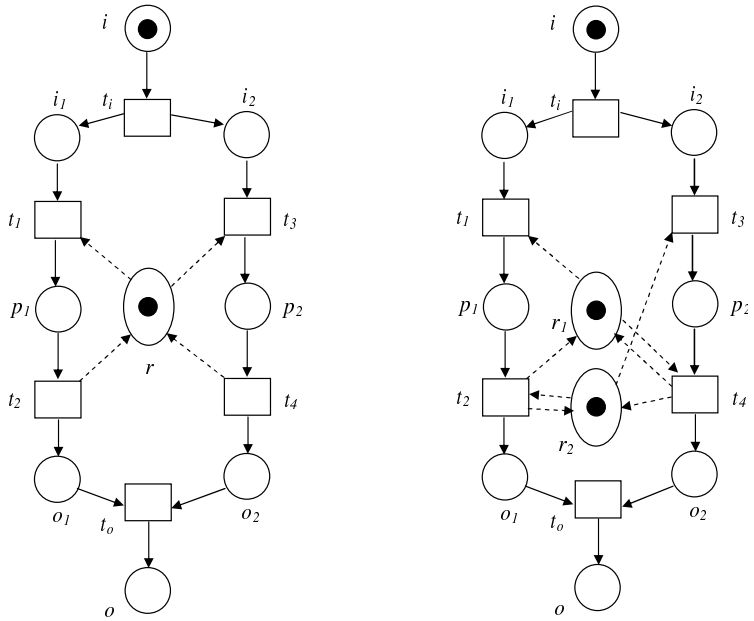
Заметим, что в утверждении Леммы 1 нельзя заменить "$r \in \mathbf{res}(N)$" на "N (r)-бездефектна", поскольку (r)-бездефектная сеть не обязательно $(r + \delta)$-бездефектна. Для (r)-бездефектности выполняется более слабое свойство:

Лемма 2. *Если сеть N (r)-бездефектна и $c|r_1, c|r_2 \in \mathcal{R}(N, i|r)$, то $r_1 \not< r_2$ и $r_1 \not> r_2$.*

Доказательство. Аналогично доказательству предыдущей леммы. Предположим противное: $r_1 < r_2$ и, следовательно, $r_2 = r_1 + \delta_1$ при $\delta_1 \neq \emptyset$.

Имеем $i|r \to c|r_1 \to o|r$ и $i|r \to c|r_2 = c|r_1 + \delta_1 \to o|r$, но тогда $c|r_1 + \delta_1 \to o|r + \delta_1$ — противоречит (r)-бездефектности. □

Из конечности множеств попарно несравнимых векторов над $Nat^{|P_r|}$ имеем:



a) бездефектная композиция

b) возможен тупик

Рис. 2. Два примера композиции RCWF-сетей

Следствие 1. *Если сеть N (r)-бездефектна, то $\mathcal{R}(N, i|r)$ конечно.*

Важным свойством является невозможность изменения ресурса при помощи цикла:

Утверждение 1. *Если сеть N бездефектна, $r \in \mathbf{res}(N)$, $c|r_1 \in \mathcal{R}(N, i|r)$ и $c|r_2 \in \mathcal{R}(N, c|r_1)$, то $r_1 = r_2$.*

Доказательство. Непосредственно из Леммы 1. □

Бездефектная сеть производит только фиксированные трансформации ресурсов:

Утверждение 2. *Если сеть N бездефектна, $r \in \mathbf{res}(N)$, $c|r' \in \mathcal{R}(N, i|r)$ и $u \in \mathcal{M}(P_r)$, то для любого $c|v \in \mathcal{R}(N, i|r + u)$ выполняется $v = r' + u$.*

Доказательство. Предположим противное. Тогда $c|v, c|r' + u \in \mathcal{R}(N, i|r + u)$ при $v \neq r' + u$, что противоречит Лемме 1. □

5. Композиции RCWF-сетей

Сети с одним и тем же множеством ресурсов могут быть соединены параллельно:

Определение 2. *Пусть N_1 и N_2 — RCWF-сети, такие что*

- $N_1 = \big((P_c)_1, P_r, T_1, (F_c)_1, (F_r)_1, i_1, o_1\big)$ *и*
- $N_2 = \big((P_c)_2, P_r, T_2, (F_c)_2, (F_r)_2, i_2, o_2\big)$.

Параллельной композицией N_1 *и* N_2 *(обозначается $N = N_1 \| N_2$) называется RCWF-сеть $N = (P_c, P_r, T, F_c, F_r, i, o)$, такая что*

- $P_c =_{def} (P_c)_1 \cup (P_c)_2 \cup \{i, o\}$,
- $T =_{def} T_1 \cup T_2 \cup \{t_i, t_o\}$,
- $F_c =_{def} (F_c)_1 \cup (F_c)_2 \cup \{(i, t_i), (t_i, i_1), (t_i, i_2), (t_o, o), (o_1, t_o), (o_2, t_o)\}$,
- $F_r =_{def} (F_r)_1 \cup (F_r)_2$.

Другими словами, мы соединяем два процесса параллельно, добавляя новые общие начальную и финальную позиции.

Примеры композиции простых сетей приведены на Рис. 2. В случае Рис. 2(a) обе подсети имеют одинаковый минимальный бездефектный ресурс r, и их композиция также бездефектна с этим ресурсом. Случай Рис. 2(b) более сложный. Заметим, что $r_1 + r_2$ является минимальным бездефектным ресурсом для обеих подсетей, однако их композиция не является $(r_1 + r_2)$-бездефектной из-за тупика $p_1 + p_2|\emptyset$, достижимого от $i|r_1 + r_2$. Любой больший ресурс уже бездефектен.

Таким образом, бездефектность ресурса для подсетей не обязательно означает его бездефектность для их композиции (как можно было бы ожидать, принимая во внимание консервативность ресурсных трансформаций в RCWF-сети). Параллелизм может порождать дополнительные тупики и блокировки. Однако можно обнаружить достаточно простое аддитивное замыкание:

Теорема 1. *Если N_1 и N_2 бездефектны, то и $N_1 \| N_2$ бездефектна, а кроме того:*

1. $r_1 \in \mathbf{res}(N_1), r_2 \in \mathbf{res}(N_2) \;\Rightarrow\; r_1 + r_2 \in \mathbf{res}(N_1 \| N_2)$;
2. $r \in \mathbf{res}(N_1 \| N_2) \;\Rightarrow\; r \in \mathbf{res}(N_1)$;
3. $r \in \mathbf{mres}(N_1 \| N_2) \;\Rightarrow\; \exists r_1 \in \mathbf{res}(N_1) : r \leq r_1$.

Доказательство. Бездефектность и первое утверждение следуют из Утв. 2. Заметим, что в данном случае подсети N_1 и N_2 могут работать независимо друг от друга, без вмешательства в "чужую" часть общего ресурса.

Второе утверждение также достаточно очевидно. Заметим, что, поскольку ресурс бездефектен для параллельной композиции, его должно быть достаточно для поддержки системных исполнений вида $i|r \to i_1 + i_2|r \to o_1 + i_2|r \to o_1 + o_2|r$.

Третье утверждение является тривиальным следствием второго. □

Из первого утверждения Теоремы 1 вытекает

Следствие 2. *Если сети N_1 и N_2 бездефектны, $r_1 \in \mathbf{mres}(N_1)$ и $r_2 \in \mathbf{mres}(N_2)$, то существует $r \in \mathbf{mres}(N_1 \| N_2)$, такое, что $r \leq r_1 + r_2$.*

Таким образом, для обнаружения какого-нибудь минимального ресурса r достаточно перебрать конечное число ресурсов, не превосходящих $r_1 + r_2$. Для каждого кандидата $r' \leq r_1 + r_2$ множество $\mathcal{R}(N, i|r')$ конечно (Следствие 1) и может быть построено за конечное число шагов. Заметим, что мы не доказали возможность обнаружения ВСЕХ элементов $\mathbf{mres}(N_1 \| N_2)$ при помощи данного метода. Тем не менее, вычисленное подмножество всегда не пусто и нетривиально.

Таким образом, проблема вычисления $\mathbf{mres}(N)$ может быть частично сведена к той же проблеме для подсетей, соединенных параллельно. В большинстве случаев процесс декомпозиции завершается последовательными потоками работ, имеющими очень простые множества бездефектных и минимально-бездефектных ресурсов.

6. Обеспечение бездефектности

В данной главе мы рассмотрим ресурс r, бездефектный для обеих подсетей, но не бездефектный для их параллельной композиции (как, например, ресурс $r_1 + r_2$ на Рис. 2(b)). Заметим, что такому ресурсу всегда соответствует непустое множество "хороших" исполнений процесса (как минимум два: $i|r \to i_1 + i_2|r \to o_1 + i_2|r \to o_1 + o_2|r$ и $i|r \to i_1 + i_2|r \to i_1 + o_2|r \to o_1 + o_2|r$). Следовательно, он не является совершенно бесполезным, и было бы интересно разработать такую стратегию управления процессом (или трансформацию сети), чтобы она разрешала все "хорошие" варианты исполнения и запрещала все "плохие". И, конечно, важно, чтобы при этом начальный ресурс не был увеличен.

Возможны два варианта нежелательного (некорректно завершаемого) поведения сети Петри — это тупик (deadlock) и динамический тупик (livelock).

Достижимая разметка $c|r$ называется *тупиковым состоянием*, если $c \neq o$ и не существует перехода $t \in T$ такого, что $c|r \xrightarrow{t} c'|r'$ для некоторых c', r'. Конечное множество L достижимых разметок называется *динамическим тупиком*, если: (1) $|L| > 1$; (2) для любых $c|r, c'|r' \in L$ найдется конечная последовательность переходов $\sigma \in T^*$ такая, что $c|r \xrightarrow{\sigma} c'|r'$; (3) для любых $c|r \in L$ и $t \in T$ таких, что $c|r \xrightarrow{t} c''|r''$, выполняется $c''|r'' \in L$. *Динамически тупиковым состоянием* называется разметка, принадлежащая какому-нибудь динамическому тупику. Через $D(N, i|r)$ обозначим множество всех тупиковых и динамически тупиковых состояний сети $(N, i|r)$.

Теорема 2. *Если* $N = N_1 \| N_2$ *и* $r \in \mathbf{res}(N_1) \cap \mathbf{res}(N_2)$, *то* $(N, i|r)$ *ограничена.*

Доказательство. Из второго утверждения Факта 2 следует, что множества управляющих разметок подсетей N_1 и N_2 конечны. Очевидно, что множество достижимых управляющих разметок N является подмножеством декартова произведения этих двух конечных множеств, следовательно, оно тоже конечно.

Рассмотрим разметки из $\mathcal{R}(N, i|r)$. Предположим противное — что это множество бесконечно. Тогда из ограниченности управляющей подсети следует существование некоего управляющего цикла, строго увеличивающего ресурс: $i|r \to c_1 + c_2|r' \xrightarrow{\sigma} c_1 + c_2|r' + r''$ при $c_1 \in \mathcal{M}((P_c)_1), c_2 \in \mathcal{M}((P_c)_2), \sigma \in T^*$ и $r'' \neq \emptyset$.

Вспомним, что $T = T_1 \cup T_2$, и обозначим через σ_1 и σ_2 наибольшие подпоследовательности σ такие, что $\sigma_1 \in (T_1)^*$ и $\sigma_2 \in (T_2)^*$. Очевидно, что σ_1 и σ_2 являются управляющими циклами в N_1 и N_2 соответственно.

Из Утверждения 1 следует, что ни σ_1, ни σ_2 не могут изменить ресурс, следовательно, и их композиция тоже не сможет этого сделать — противоречие. □

Поскольку $D(N, i|r) \subseteq \mathcal{R}(N, i|r)$, мы имеем:

Следствие 3. *Если* $N = N_1 \| N_2$ *и* $r \in \mathbf{res}(N_1) \cap \mathbf{res}(N_2)$, *то* $D(N, i|r)$ *конечно.*

Итак, множество тупиковых состояний вычислимо за конечное время простым перебором конечного множества достижимости сети. Таким образом, "наивная" стратегия управления блокировками может состоять в предварительном обнаружении всех возможных тупиков и последующем наблюдении за текущим поведением процесса с целью недопущения неверного "последнего шага".

6.1. Безопасные сети

Интересным частным случаем являются безопасные потоки работ, то есть RCWF-сети с безопасными управляющими подсетями (где ни одна из управляющих позиций не может накопить более одной фишки). Такое ограничение не слишком сужает класс моделируемых систем, поскольку всякая ограниченная сеть бисимулярна (в слабом смысле) некоторой безопасной. Заметим, что сеть на Рис. 2(b) безопасна, и, тем не менее, содержит тупик. В безопасной RCWF-сети все управляющие дуги *ординарны*: $F_c(x, y) \leq 1$ для любых x и y.

Следующая трансформация безопасной сети избавляет её от всех тупиков:

Определение 3. *Пусть N_1 и N_2 — безопасные RCWF-сети с одним и тем же множеством P_r ресурсных позиций, и пусть $r \in \mathcal{M}(P_r)$ — такой ресурс, что $r \in \mathbf{res}(N_1)$ и $r \in \mathbf{res}(N_2)$.*

Пусть $N = N_1 \| N_2 = (P_c, P_r, T, F_c, F_r, i, o)$ — не являющаяся (r)-бездефектной параллельная композиция N_1 и N_2.

Через $D_c(N, i|r)$ обозначим множество всех различных управляющих частей элементов из $D(N, i|r)$, и пусть $Z = |D_c(N, i|r)|$ (очевидно, что $Z > 0$).

Сеть $(N_a, i|r + v)$, где $N_a = (P_c, P_r \cup V, T, F_c, F_r \cup F_{in} \cup F_{out}, i, o)$, назовём управляемой системой *сети $(N, i|r)$, если*

- *$V = \{v_k | k \in \overline{1, Z}\}$ — множество "холдеров" (удерживающих позиций), и их количество равно числу всех возможных тупиковых состояний сети N;*
- *F_{in} — входные дуги холдеров такие, что $F_{in} = \{(v_k, t) \mid F_c(t, p) = 1$ для некоторого $p \in d_k$, где d_k — k-ый элемент $D_c(N, i|r)\}$;*
- *F_{out} — выходные дуги холдеров такие, что $F_{out} = \{(t, v_k) \mid F_c(p, t) = 1$ для некоторого $p \in d_k$, где d_k — k-ый элемент $D_c(N, i|r)\}$;*
- *$v = (|d_1|-1)v_1+(|d_2|-1)v_2+\ldots+(|d_Z|-1)v_Z$, где d_k — k-ый элемент $D_c(N, i|r)$.*

Основная идея заключается в том, что нам необходимо избегать последовательного срабатывания переходов, ведущих к "тупиковым" позициям. Для этого вводится удерживающая позиция, ресурс (фишка) в которой разрешает срабатывать переходам, ведущим только к одной позиции из набора позиций данного тупика. Следует обратить внимание на то, что холдер не должен препятствовать срабатыванию переходов, потребляющих не критический ресурс (т.к. его обнуление не приводит к тупику). Вернуть фишку в холдер необходимо сразу, как только фишка покинет "тупиковую" позицию. На Рис. 3 показана реализация этой идеи для сети из примера на Рис. 2.

Теорема 3. *Пусть N_1 и N_2 — бездефектные безопасные RCWF-сети с одним и тем же множеством P_r ресурсных позиций, и пусть $r \in \mathcal{M}(P_r)$ — такой ресурс, что $r \in \mathbf{res}(N_1)$ и $r \in \mathbf{res}(N_2)$.*

Пусть $N = N_1 \| N_2 = (P_c, P_r, T, F_c, F_r, i, o)$ — не являющаяся (r)-бездефектной параллельная композиция N_1 и N_2.

Пусть $(N_a, i|r + v)$ — управляемая система размеченной сети $(N, i|r)$. Тогда $(N_a, i|r + v)$ — $(r + v)$-бездефектна.

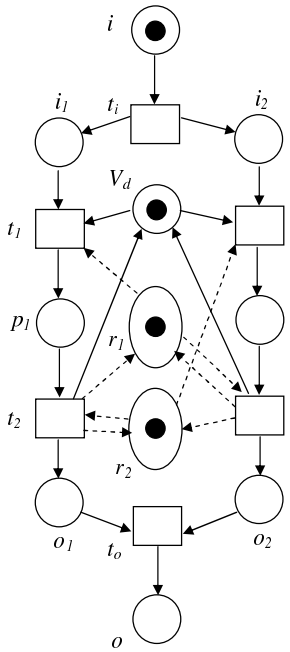


Рис. 3. Пример композиции RCWF-сетей с управляемыми блокировками

Доказательство. Очевидно, что в модифицированной сети не достижим ни один из тупиков исходной (по построению).

Теперь докажем, что не появилось новых тупиков. Рассмотрим некоторое тупиковое состояние $c_1 + c_2|r$ исходной сети. Из третьего утверждения Факта 2 и свойства безопасности сети получим, что ни одно из управляющих состояний сети N_1, кроме c_1, не может иметь того же самого или большего количества фишек во всех позициях из c_1. Аналогично — для сети N_2 и позиций из c_2. Следовательно, $|c_1+c_2|-1 = |c_1|+|c_2|-1$ фишек будет достаточно для всех управляющих состояний композиции, кроме данного конкретного тупика. Таким образом, соответствующий холдер не вносит никаких дополнительных ограничений в поведение сети. □

7. Заключение

Мы представили два метода борьбы с блокировками для ограниченного ресурса. Первый из них применим к любой паре бездефектных потоков работ, но требует дополнительного (внешнего) контроля сети во время её исполнения. Второй использует конструкции самой сети Петри (добавляются новые позиции и дуги), но применим только лишь к безопасным сетям. Представленная во втором случае техника близка к технике, используемой при управлении гибкими производственными системами (FMS — Flexible Manufacturing Systems) [12]. Однако существенным отличием нашего случая является возможность параллельного функционирования внутри подсетей (в FMS каждая “подсеть” представляет собой простой последовательный автомат).

Одним из возможных направлений дальнейших исследований является проблема точного вычисления множества $\mathbf{mres}(N)$. Мы полагаем, что представленный в

данной работе подход может быть применим для её решения (по крайней мере, для многих важных подклассов RCWF-сетей). В частности, мы планируем исследовать структурированные потоки работ [1, 11], которые могут быть получены из базовых примитивов через последовательность алгебраических операций, таких как параллельная и последовательная композиции. Ещё один интересный метод композиции ресурсов, основанный на алгебре мультимножеств, был предложен в [14].

Список литературы

1. *Ван дер Аалст В., ван Хей К.* Управление потоками работ: модели, методы и системы. М.: Научный мир, 2007. (English transl.: *van der Aalst W., van Hee K.* Workflow Management: Models, Methods and Systems. MIT Press, 2002.)

2. *Башкин В. А.* Сети активных ресурсов // Моделирование и анализ информационных систем. 2007. Т. 14. № 4. С. 13–19.

3. *Башкин В. А.* Формализация семантики систем с ненадежными агентами при помощи сетей активных ресурсов // Программирование. 2010. №4. С.3–15. (English transl.: *Bashkin V. A.* Formalization of semantics of systems with unreliable agents by means of nets of active resources // Programming and Computer Software. 2010. **36**(4). P. 187–196.)

4. *Van der Aalst W. M. P.* The Application of Petri Nets to Workflow Management // The Journal of Circuits, Systems and Computers. 1998. **8**(1). P. 21–66.

5. *Van der Aalst W.M.P., van Hee K.M., Hofstede A.H.M., Sidorova N., Verbeek H.M.W., Voorhoeve M., Wynn M.T.* Soundness of workflow nets: classification, decidability, and analysis // Formal Aspects of Computing. 2011. **23**(3). P. 333–363.

6. *Barkaoui K., Petrucci L.* Structural Analysis of Workflow Nets with Shared Resources // Proc. of Workflow Management: Net-based Concepts, Models, Techniques and Tools (WFM98). Computing Science Reports. Eindhoven University of Technology. 1998. Vol. 98/7. P. 82–95.

7. *Barkaoui K., Ben Ayed R., Sbaï Z.* Workflow Soundness Verification based on Structure Theory of Petri Nets // International Journal of Computing and Information Sciences. 2007. **5**(1). P. 51–61.

8. *Bashkin V. A., Lomazova I. A.* Petri nets and resource bisimulation // Fundamenta Informaticae. 2003. Vol. 55. No. 2. P. 101–114.

9. *Bashkin V. A., Lomazova I. A.* Resource equivalence in workflow nets // Proc. of Concurrency, Specification and Programming (CS&P'2006). Humboldt Universitat zu Berlin, 2006. Vol. 1. P. 80–91.

10. *Bashkin V. A., Lomazova I. A.* Soundness of Workflow Nets with an Unbounded Resource is Decidable // Joint Proc. of Petri Nets and Software Engineering (PNSE'13) and Modeling and Business Environments (ModBE'13). 2013. Vol. 989 of CEUR. P. 61–75.

11. *Chrząstowski-Wachtel P.* Sound Markings in Structured Nets // Proc. of Concurrency, Specification and Programming (CS&P'2005). Warsaw University, 2005. P. 71–85.

12. *Ezpeleta J., Colom J.-M., Martinez J.* A Petri Net Based Deadlock Prevention Policy for Flexible Manufacturing Systems // IEEE Transactions on Robotics and Automation. 1995. **11**(2). P. 173–184.

13. *Van Hee K., Serebrenik A., Sidorova N., Voorhoeve M.* Soundness of Resource-Constrained Workflow Nets // Proc. of ICATPN 2005. Lecture Notes in Computer Science. 2005. Vol. 3536. P. 250–267.

14. *Lomazova I. A., Romanov I. V.* Analyzing Compatibility of Services via Resource Conformance // Fundamenta Informaticae. 2013. Vol. 128. No. 1–2. P. 129–141.

15. *Sidorova N., Stahl C.* Soundness for resource-contrained workflow nets is decidable // IEEE Transactions on Systems, Man, and Cybernetics: Systems. 2013. **43**(3). P. 724–729.

Controllable Deadlocks in Parallel Resource-Constrained Workflows

Bashkin V.A., Panfilova N.Yu.

P.G. Demidov Yaroslavl State University,
Sovetskaya str., 14, Yaroslavl, 150000, Russia

Keywords: workflow, resource, soundness, parallel composition, deadlock, verification

We study the verification of the soundness property for workflow nets extended with resources. A workflow is sound if it terminates properly (no deadlocks and livelocks are possible). A class of resource-constrained workflow nets (RCWF-nets) is considered, where resources can be used by a process instance, but cannot be created or spent. Two sound RCWF-nets using the same set of resources can be put in parallel. This parallel composition may in some cases produce additional deadlocks. A problem of deadlock avoidance in parallel workflows is studied, some methods of deadlock search and control are presented.

Сведения об авторах:
Башкин Владимир Анатольевич,
Ярославский государственный университет им. П. Г. Демидова,
докт. физ.-мат. наук, доцент;
Панфилова Надежда Юрьевна,
Ярославский государственный университет им. П. Г. Демидова,
магистрант.