

А.А. Тимочко

Фирма «Kreditech», Гамбург, Германия

ПРЕДСТАВЛЕНИЕ ДИНАМИЧЕСКИХ ВЗАИМОДЕЙСТВУЮЩИХ ПРОЦЕССОВ В ПРОСТРАНСТВЕ СОСТОЯНИЙ НЕЧЕТКОЙ РАСКРАШЕННОЙ СЕТИ ПЕТРИ ДЛЯ ВЕРИФИКАЦИИ ПРОГРАММНОГО ПРОДУКТА НЕЧЕТКОЙ ЛОГИЧЕСКОЙ СИСТЕМЫ КЛАССИФИКАЦИИ

Процесс верификации программного обеспечения системы классификации воздушных объектов, при контроле воздушного пространства в нечеткой постановке, рассматривается как процесс исследования динамических взаимодействующих процессов. Утверждается, что традиционные подходы не позволяют с заданным качеством верифицировать программное обеспечение при решении задачи классификации воздушных объектов в автоматизированных системах контроля воздушного пространства в заданной постановке. Цель статьи – формализация представления динамических взаимодействующих процессов в пространстве состояний нечеткой раскрашенной сети Петри для верификации программного обеспечения нечеткой логической системы классификации воздушных объектов в автоматизированных системах контроля воздушного пространства. Проанализированы основные работы в области теории и практики применения сетей Петри, использованные для исследования динамических взаимодействующих процессов в общем, и непосредственно для верификации программного обеспечения различных систем. Выдвинуты требования к сетям Петри для решения задач данного класса. Определены условия разрешенности при представлении взаимодействующих процессов для различных случаев, которые отличаются количеством входных и (или) выходных переходов. Для описания компонентов модели верификации программного обеспечения системы классификации воздушных объектов, представленных логикой предикатов, использована нечеткая раскрашенная сеть Петри (для логических операций И, ИЛИ, НЕ) и соответствующие функции принадлежности. Разработанный подход по представлению динамических взаимодействующих процессов в пространстве состояний нечеткой раскрашенной сетью Петри для верификации программного продукта нечеткой логической системы классификации является основой для разработки базы нечетких продукционных правил.

Ключевые слова: программное обеспечение, верификация, классификация, воздушный объект, взаимодействующий процесс, нечеткая раскрашенная сеть Петри.

Введение

Постановка проблемы. Процесс верификации программного обеспечения (ПО) нечеткой логической системы классификации воздушных объектов (ВО) при контроле воздушного пространства в общем случае можно рассматривать как процесс исследования динамических взаимодействующих процессов.

С одной стороны, к настоящему моменту в основном сформулированы основные положения теории и практики применения сетей Петри для исследования динамических взаимодействующих процессов. С другой стороны, известны основные подходы для верификации ПО различных систем.

Однако традиционные подходы не позволяют с заданным качеством верифицировать программное обеспечение при решении задачи классификации воздушных объектов в автоматизированных системах контроля воздушного пространства.

Таким образом, существует необходимость разработки новых методов представления динамических взаимодействующих процессов в пространстве состояний нечеткой раскрашенной сети Петри для верификации программного продукта нечеткой логической системы классификации ВО в автоматизированных системах контроля воздушного пространства.

зированных системах контроля воздушного пространства.

Целью статьи является формализация представления динамических взаимодействующих процессов в пространстве состояний нечеткой раскрашенной сети Петри для верификации программного продукта нечеткой логической системы классификации ВО в автоматизированных системах контроля воздушного пространства.

Анализ литературы. Рассмотрим основные работы в области теории и практики применения сетей Петри. При этом внимание сосредоточим как в целом для исследования динамических взаимодействующих процессов, так и непосредственно для верификации ПО различных систем. Особый интерес, естественно, вызывают нечеткие логические системы классификации ВО в процессе контроля воздушного пространства. Так, в работе [1] предложены подходы к построению моделей на основе нечеткой логики и сетей Петри, что позволяет исследовать нечеткие правила в базах знаний. Нечеткие сети Петри, описанные в работе, представляются кортежем:

$$\tilde{S}(f) = \langle \tilde{P}, \tilde{T}, F, I, O, M \rangle, \quad (1)$$

где $\tilde{P} = \{\tilde{p}_j\}$ – конечное множество нечетких позиций \tilde{p}_j , которым ставятся в соответствие функции принадлежности $\mu_{\tilde{p}_j}(k)$; k – некоторая переменная, определяющая значение функции $\tilde{T} = \{\tilde{t}_i\}$ – конечное множество нечетких переходов \tilde{t}_i , которым ставятся в соответствие функции принадлежности $\mu_{\tilde{t}_i}(k)$; k – некоторая переменная, определяющая значение функции $\mu_{\tilde{t}_i}(k)$, $i = \overline{1, n}$, $T \neq \emptyset$, $|\tilde{T}| = n$; функция инцидентностей

$$F : (P \times T) \cup (T \times P); \quad (2)$$

входная $I : (P \times T)$ и выходная $O : (T \times P)$ функции инцидентностей.

Маркирование позиций определяется функцией $M : (\tilde{P}) \rightarrow N$.

Предложенная структура позволяет представлять нечеткие процессы и динамику их взаимодействия. Однако в ней не учитывается множество параметров, показателей и характеристик, без которых сложно представить реальные процессы практических реализаций. Кроме того, данная структура ограничивает возможность задания показателей нечеткости маркировки и компонент функции инцидентности.

Для сокращения размерности модели и расширения возможностей моделирования процессов и систем в работах [2–3] предложено моделирование на основе раскрашенных сетей Петри.

В работе [4] исследованы раскрашенные сети Петри и их расширения в задачах автоматизации проектов создания систем. Доказано, что раскрашенные нечеткие сети Петри улучшают качество и гибкость анализа объектов исследования. Структура, представленная выражениями (1) и (2) по существу дополняется функцией цвета:

$$\tilde{S}(f) = \langle \tilde{P}, \tilde{T}, F, I, O, C, V, K, M \rangle, \quad (3)$$

где V – условие срабатывания перехода с учетом функции цвета, K – емкость меток в позициях.

В работе [5] определена нечеткая сеть Петри на основе использования значений локальной функции принадлежности позиций и переходов и ограничений на ее значение. В работе [6] эта модель расширена путем постановки в соответствие маркеру значения функции принадлежности. Этот прием позволяет собственно осуществлять ограниченный нечеткий логический вывод.

В работах [7–8] разработана нечеткая сеть Петри, определенная как сеть высокого уровня. По существу данная сеть является раскрашенной сетью Петри. В ней метка позиции определена значением функции принадлежности, для позиций \tilde{P} , перехо-

дов \tilde{T} определены функции принадлежности $\mu(x)$, а для входных и выходных инцидентий перехода определены нечеткие метки $\langle b_1, \dots, b_n \rangle$, соответственно, которые и определяют тип нечетких входного и выходного объектов, а также матрицу отношений $F_{\tilde{A} \rightarrow \tilde{B}}$ в решении

$\tilde{B}'(y) = \max(\tilde{A}'(x) \text{t} F_{\tilde{A} \rightarrow \tilde{B}}(x, y))$ на основе прямого логического вывода

if V is A then U is B
V is A'

U is B'

для случая использования упрощения Mamdani оператора Zadeh.

В работах [9–10] рассматриваются некоторые частные и специфические вопросы построения сетевых моделей и процедур логического вывода на нечетких сетях Петри. Анализ работ показал, что ограничение учета нечеткости преимущественно позициями и переходами уменьшает возможности моделирования и практическую значимость получаемых результатов. В работе [11] исследованы вопросы устойчивости и эффективности решений задач нечеткого управления в системах с использованием сетей Петри. Основные результаты работы показывают, что анализ поведенческих свойств моделей, отображающих взаимодействующие динамические нечеткие процессы, во многом является определяющим в практических реализациях.

В работах [12–13] исследуются решения поведенческих свойств фрагментов баз знаний (БЗ) с использованием нечетких сетей Петри. В частности, рассмотрены возможности выявления свойств конфликтности, наличия ошибок и избыточности в БЗ.

В работе [14] приведены некоторые представления о построении инструментальных средств моделирования и анализа структурных свойств нечеткой модели, описывающей процессы нечеткой системы управления. БЗ представлена продукционными правилами. Рассматривается возможность выявления конфликтов и избыточности внутри правил. Это требует запуска определенных процедур устранения некорректного представления процессов путем удаления избыточности и развязывания конфликта запрещающими дугами. Устранение противоречивости также возможно путем корректировки пространства состояний модели и удаления соответствующих меток. В последующем осуществляется соответствующая коррекция антецедентов правил БЗ. Анализ показал, что предлагаемый подход находится во многом на этапе постановки, носит узкоспециальный и иллюстративный характер без детальной формализации.

В работе [15] приведены различные варианты временных сетей Петри. Оцениваются недостатки и

достоинства с точки зрения применения их для разработки программного обеспечения совместно с диаграммами UML в задачах распределенного взаимодействия в условиях реального времени.

В работе [16] рассматривается моделирование параллельных процессов в вычислительных системах с помощью аппарата сетей Петри. Предлагается проводить построение модели параллельных процессов на основе интерпретаций сетей Петри.

В работе [17] рассматриваются проблемы формального доказательства корректности требований к системам дистанционного контроля и управления. Предлагается методика представления и анализа требований с помощью UML-диаграмм и их верификации с помощью цветных иерархических сетей Петри и дается её описание.

В работе [18] рассматривается применение иерархических сетей Петри при разработке программного обеспечения с использованием диаграмм UML. Приводится построение диаграммы прецедентов по словесному описанию системы. Строится сеть Петри, отвечающая за поведенческие аспекты работы. Полученная сеть тестируется и преобразуется в диаграмму деятельности, после чего происходит построение диаграммы классов и объектов.

В работе [19] разработаны методы и алгоритмы верификации моделей распределенных систем, базирующихся на раскрашенных сетях Петри. Определены развертки раскрашенных сетей Петри. Для раскрашенных сетей Петри разработан метод проверки моделей с использованием разверток.

Таким образом, имеет место противоречие. С одной стороны, существующие подходы к построению раскрашенных сетей Петри и раскрашенных нечетких сетей Петри, обладая дополнительными возможностями, не свободны от приведенных недостатков. Это, естественно, не позволяет их применять в современных разработках без дальнейшего развития и новых принципиальных решений. С другой стороны, отсутствуют методы, позволяющие осуществлять непосредственную ПО нечеткой логической системы классификации воздушных объектов в процессе контроля воздушного пространства.

Поэтому представляется необходимым использование новых классов расширенных нечетких раскрашенных сетей Петри, свободных от указанных выше недостатков и позволяющих осуществлять верификацию программного обеспечения нечеткой логической системы классификации.

Изложение основного материала

Итак, для решения задач данного класса сети Петри должны характеризоваться следующим:

– возможностью создания нечетких сетевых моделей. Они характеризуются естественной интерпретацией, простотой описания и моделирования взаимодействующих нечетких динамических

процессов, представленных на множестве отношений «условие-действие» с учетом множества реальных параметров, характеристик, показателей и ограничений предметной области;

– адаптацией к классам задач и предметной области при решении комплекса поставленных задач верификации программного обеспечения нечеткой логической системы классификации;

– решением комплекса поставленных задач как единой проблемы создания моделей, критериев, методов, интеллектуальных технологий и эффективных инструментальных средств.

Представление взаимодействующих процессов рассматривается для случаев, когда [18]:

– процесс выполняется при наличии ровно одного входного и ровно одного выходного условий

$$(\exists \tilde{t})_i \in \tilde{T} (|\{\tilde{p}_i(\text{in})\}| = |\{\tilde{p}_i(\text{out})\}| = 1); \quad (4)$$

– процесс выполняется при наличии нескольких, неравных одному, входных и ровно одного выходного условий

$$(\exists \tilde{t})_i \in \tilde{T} (|\{\tilde{p}_i(\text{in})\}| > 1) \text{ and } (|\{\tilde{p}_i(\text{out})\}| = 1); \quad (5)$$

– процесс выполняется при наличии ровно одного входного и нескольких, неравных одному, выходных условий

$$(\exists \tilde{t})_i \in \tilde{T} (|\{\tilde{p}_i(\text{in})\}| = 1) \text{ and } (|\{\tilde{p}_i(\text{out})\}| > 1); \quad (6)$$

– некоторое условие выполнения процесса имеет несколько, не равных одному, входных процессов и ровно один выходной процесс

$$(\exists \tilde{p})_i \in \tilde{P} (|\{\tilde{t}_j(\text{in})\}| > 1) \text{ and } (|\{\tilde{t}_j(\text{out})\}| = 1); \quad (7)$$

– некоторое условие выполнения процесса имеет ровно один входной процесс и несколько, не равных одному, выходных процессов

$$(\exists \tilde{p})_i \in \tilde{P} (|\{\tilde{t}_j(\text{in})\}| = 1) \text{ and } (|\{\tilde{t}_j(\text{out})\}| > 1); \quad (8)$$

– некоторое условие имеет только ровно один выходной процесс

$$(\exists \tilde{p})_i \in \tilde{P} (|\{\tilde{t}_j(\text{in})\}| = 1) \text{ and } (|\{\tilde{t}_j(\text{out})\}| = 1); \quad (9)$$

– некоторое условие имеет только ровно один входной процесс

$$(\exists \tilde{p})_i \in \tilde{P} (|\{\tilde{t}_j(\text{in})\}| = 1) \text{ and } (|\{\tilde{t}_j(\text{out})\}| = 0); \quad (10)$$

– процесс выполняется при наличии нескольких, не равных одному, входных и нескольких, не равных одному, выходных условий

$$(\exists \tilde{t})_i \in \tilde{T} (|\{\tilde{p}_i(\text{in})\}| > 1) \text{ and } (|\{\tilde{p}_i(\text{out})\}| > 1); \quad (11)$$

– некоторое условие выполнения процесса имеет несколько, не равных одному, входных процессов и несколько, не равных одному, выходных процессов

$$(\exists \tilde{p})_i \in \tilde{P} (|\{\tilde{t}_j(\text{in})\}| > 1) \text{ and } (|\{\tilde{t}_j(\text{out})\}| > 1); \quad (12)$$

где $\{\tilde{t}_j(\text{in})\}$ – множество входных переходов позиции \tilde{p}_j ; $\{\tilde{t}_j(\text{out})\}$ – множество выходных переходов позиции \tilde{p}_j .

Выражение (11) может быть представлено как последовательное соединение выражений (5) и (6), а выражение (12) – как последовательное соединение выражений (7) и (8) [18].

Для каждого из выражений (4–8) определяются условия разрешенности переходов и условия маркирования позиций моделей.

Выражения (4–6) отличаются только количеством входных $\{\tilde{p}_i(\text{in})\}$ и (или) выходных позиций $\{\tilde{p}_i(\text{out})\}$ рассматриваемого перехода \tilde{t}_i . Поэтому для них справедливы соответствующие условия

$$((\tilde{t}_i \in \tilde{T} : \mu_{\tilde{t}_i}(k_0) \geq (\mu_{\tilde{t}_i}(k_0))^*) \text{ and } ((\forall \tilde{p}_j \in \{\tilde{p}_i(\text{in})\} | p_j : \mu_{\tilde{p}_j}(k_0) \geq \mu_{\tilde{p}_j}(k_0)^*) \text{ and } (\forall \tilde{M}(\tilde{p}_j) \in \tilde{M}(f) | \tilde{M}(\tilde{p}_j) \geq 1) \text{ and } (z_{\tilde{p}_j}(k_0) \geq (z_{\tilde{p}_j}(k_0))^*) \text{ and } \text{and } (x_{ij}(k_0) \geq (x_{ij}(k_0))^*) \text{ and } L = \text{true}),$$

где $\mu_{\tilde{t}_i}(k_0)^*$, $\mu_{\tilde{p}_j}(k_0)^*$, $z_{\tilde{p}_i}(k_0)^*$, $z_{\tilde{p}_j}(k_0)^*$, $x_{ij}(k_0)^*$ – ограничения на значения соответствующих функций принадлежности; k_0 – некоторое значение переменной k , определяющее конкретное, исходя из оценок о предметной области, значение соответствующей функции принадлежности.

Учитывая, что выражения (7–10) отличаются количеством входных $\{\tilde{t}_j(\text{in})\}$ и (или) выходных $\{\tilde{t}_j(\text{out})\}$ переходов рассматриваемой позиции \tilde{p}_j , то для переходов $\{\tilde{t}_i\}$, входящих в соответствующее выражение из (7–10), также справедливы приведенные выше условия их разрешенности.

Взаимодействующие динамические нечеткие процессы, соответствующие верификации ПО нечеткой логической системы классификации воздушных объектов в процессе контроля воздушного пространства, могут быть формально представлены в виде следующих аналитических представлений:

а) логики предикатов, содержащих логические, включая нечеткие, операции – И, ИЛИ, НЕ, их производные и логические функции;

б) графическое представление, например, в виде граф-схем алгоритмов, включая нечеткие правила продукций на основе $\frac{\text{if}}{\text{then}}$ отношений, содержащих в общем случае и нечеткие представления.

Таким образом, при описании взаимодействующих динамических нечетких процессов, представленных логикой предикатов, необходимо, по крайней мере, представить с использованием нечеткой раскрашенной сети Петри логические операции И, ИЛИ, НЕ и функциями принадлежности $\mu_i, i \in I$ компонент модели верификации ПО нечеткой логической системы классификации ВО.

Операция $\text{and } a_i, i \in I$ может быть представлена на выражением (5) [18]. Данное положение основано на свойстве разрешенности перехода \tilde{t}_i выражения (5) при маркировании всех входных позиций $\tilde{p}_j \in \{\tilde{p}_i(\text{in})\}$ фрагмента. Операция $\text{or } a_i, i \in I$ может быть представлена фрагментом модели (7). Данное положение основано на свойстве маркирования позиции p_j фрагмента (7) при выполнении, по крайней мере, одного перехода $\tilde{t}_i \in \{\tilde{t}_j(\text{in})\}$ позиции p_j .

В данной интерпретации в выражении (7) возникает возможность нарушения безопасности и возникновения конфликта. Это необходимо учитывать при верификации ПО нечеткой логической системы классификации воздушных объектов на практике.

Операция НЕ может быть представлена вводом в выражение (4) ингибиторной дуги и модификации разрешенности его перехода так, что он будет разрешен при справедливости выражения [18].

$$(\exists \tilde{t}_i \in \tilde{T} | (|\{\tilde{p}_i(\text{in})\}| = |\{\tilde{p}_i(\text{out})\}| = 1) \text{ and } \text{and } (\tilde{M}(\tilde{p}_j) = 0 \text{ or } z_{\tilde{p}_j}(k_0) < (z_{\tilde{p}_j}(k_0))^*)). \quad (13)$$

Данное положение основано на свойстве ингибиторных дуг, когда разрешенность перехода \tilde{t}_i модифицированного фрагмента (4) возможна при отсутствии маркирования входной позиции $\tilde{p}_j \in \{\tilde{p}_i(\text{in})\}$ фрагмента и(или) справедливости (13).

Зададим взаимодействующие динамические нечеткие процессы, сложные процедуры в виде графического представления алгоритмов. Тогда необходимо выделить и описать следующие элементы нечеткой логической системы классификации ВО: вычислительный процесс, процесс управления, процесс принятия решений; развитие процессов по логическому условию; развитие процессов по выполнению, по крайней мере, одного из предыдущих; распараллеливание процессов; развитие процессов по окончанию всех предыдущих; начало развития процессов; получение искомого результата.

Отдельные фрагменты алгоритмов, например типа «предопределенный процесс» или вывод результатов на внешнее устройство и др., могут быть без потери адекватности их отображения представлены приведенными выше выражениями.

Основные положения, определяющие интерпретацию фрагментов модели, формулируются так [18]:

– вычислительный процесс, процесс управления, принятия решений может быть представлен на модели выражением (4);

– развитие нечетких процессов по выполнению логического условия на модели может реализовано выражением (9);

– развитие процессов по выполнению, по крайней мере, одного из предыдущих на модели может быть реализовано выражением (10);

– процедура распараллеливания процессов может быть представлена выражением (6) модели;
 – процедура развития процессов $\{P_i\}$ по окончании всех предыдущих может быть представлена выражением (5) модели;
 – процедуры начала развития процессов и получения искомого результата могут быть представлены соответственно выражениями (9) и (10).

Процедуры представления правил продукций вида $\frac{\text{if}}{\text{then}}$ определяются выражениями (4–10) модели. Если фрагменты БЗ заданы правилами продукций вида $\frac{\text{if}}{\text{then}}$, то очевидно, что правила продукций могут быть представленными выражениями модели $\tilde{S}_C(f)$, аналогично представлению взаимодействующих процессов предикатами, логическими функциями на основе логических операций.

Рассмотрим структуру правила продукции в четком представлении знаний:

$$\text{if } A \text{ and } B \text{ and } C \text{ then } D. \quad (14)$$

В вербальном изложении (14) может быть представлено следующим образом: если справедливо A и B и C , тогда выполнить действие D . Или на языке булевой логики:

$$D = \text{true}(A \text{ and } B \text{ and } C) = \text{true}. \quad (15)$$

Для выражений типа (15) уже найдены решения по представлению логических операций выражениями из (4–8) нечетких сетевых моделей (15). Можно показать справедливость соответствующих решений для правил продукций, содержащих операции ИЛИ, НЕ и их производные.

Свойства нечеткости как компонент предикатов, так и графического представления алгоритмов, а также правил продукций полностью определяются соответствующими функциями принадлежности. Аналогично формуле (14), выражение (15) в нечетком представлении может быть определено так:

$$\text{if } \tilde{A} \text{ is } \mu_{\tilde{A}}(k) \text{ and } \tilde{B} \text{ is } \mu_{\tilde{B}}(k) \text{ and } \tilde{C} \text{ is } \mu_{\tilde{C}}(k) \text{ then } \tilde{D} \text{ is } \mu_{\tilde{D}}(k). \quad (16)$$

$$\begin{aligned} \tilde{D} = \text{true} | & ((\tilde{A} \text{ and } \tilde{B} \text{ and } \tilde{C}) = \text{true}) \text{ and } ((\mu_{\tilde{A}}(k_0) \geq \\ & \geq (\mu_{\tilde{A}}(k_0))^*) \text{ and } ((\mu_{\tilde{B}}(k_0) \geq (\mu_{\tilde{B}}(k_0))^*) \text{ and} \\ & \text{and } ((\mu_{\tilde{C}}(k_0) \geq (\mu_{\tilde{C}}(k_0))^*) \text{ and } ((\mu_{\tilde{D}}(k_0) \geq \\ & \geq (\mu_{\tilde{D}}(k_0))^*), \end{aligned} \quad (17)$$

где $(\mu_{\tilde{A}}(k_0))^*$, $(\mu_{\tilde{B}}(k_0))^*$, $(\mu_{\tilde{C}}(k_0))^*$, $(\mu_{\tilde{D}}(k_0))^*$ – допустимые значения соответствующих функций принадлежности.

Функции принадлежности $\{\mu_{\tilde{P}_i}(k)\}$ множества процессов $\{\tilde{P}_i\}$, определяющие условия и действия предметной области, отображаются на множестве нечетких позиций \tilde{p}_j и нечетких переходов в пространстве состояний модели $\tilde{S}_C(f)$. Это связано с тем, что нечеткая модель $\tilde{S}_C(f)$ отображает нечеткие процессы. Например, для представления фрагмента некоторых нечетких знаний правилом продукции, которое включает нечеткие условия и нечеткие действия, можно записать в виде (16).

Выводы

1. В качестве базового математического аппарата для верификации ПО нечеткой логической системы классификации воздушных объектов в процессе контроля воздушного пространства выбраны расширенные нечеткие раскрашенные сети Петри. Данный класс сетей Петри обеспечивает снижение зависимости размерности нечеткой сетевой модели верификации от размерности динамических взаимодействующих процессов предметной области задачи классификации воздушных объектов.

2. Разработанный подход по представлению динамических взаимодействующих процессов в пространстве состояний нечеткой раскрашенной сетью Петри для верификации программного продукта нечеткой логической системы классификации ВО в процессе контроля воздушного пространства и обобщенного алгоритма реализации этих правил. Формальное представление процесса классификации воздушных объектов с использованием нечеткой логической модели позволяет учесть нестохастический и субъективный характер процесса принятия решения оператором.

3. Усилия, направленные на разработку базы нечетких производственных правил нечеткой логической системы классификации воздушных объектов в процессе контроля воздушного пространства и обобщенного алгоритма реализации этих правил, послужат основой для создания ПО нечеткой логической системы классификации ВО.

Список літератури

1. Lipp H.-P. Einsatz von zeitbewerteten Fuzzy - Petri-Netzen in Expertensystemen zur operativen Fuehrung komplexer Productionsysteme / H.-P. Lipp; Hommel, G. (Hrsg.) // Prozesrechnungssysteme 91. – Berlin/Heidelberg/New York: Springer-Verlag, 1991. – P. 103-112.
2. Jensen K. Coloured Petri nets / K. Jensen // Petri Nets: Central Models and Their Properties; W. Brauer, W. Reisig, Rosenberg (editor). – New York: Springer-Verlag, 1986. – P. 248-299.
3. Christensen S. Modular state space analysis of colored Petri Nets/Application and Theory of Petri Nets. / S. Christensen, L. Petrucci; G. De Michelis, M. Diaz (eds). – New York: Springer-Verlag, 1995. – P. 201-217.

4. Rokyta P. Electronic system design automation using high level Petri nets / P. Rokyta, W. Fengler, T. Hummel // *Workshop for Hardware Design and Petri Nets*. – Lisboa, June 22-26, 1998. – P. 129-138.
5. Cao T. Variable reasoning and analysis about uncertainty with fuzzy Petri nets / T. Cao, A. Sanderson // *Application and Theory of Petri Nets*; M.A. Marsan (ed.). – Springer-Verlag, 1993. – P. 126-145.
6. Srinivan P. Approximate reasoning with fuzzy Petri nets / P. Srinivan, D. Gracarin // *Proc. IEEE INT. Conf. on Fuzzy Systems*. – San Francisco, California, USA, 1993. – P. 396-401.
7. Scarpelli H. Fuzzy reasoning and fuzzy Petri nets / H. Scarpelli, F. Gomide // *Proc. Fifth. IFSA World Congress, Seoul, Korea, July 5-9, 1993*. – P. 1326-1329.
8. Scarpelli H. A reasoning algorithm for high-level fuzzy Petri nets / H. Scarpelli, F. Gomide, R.R. Yager // *IEEE Trans, on Fuzzy Systems*. –1996. – 4. – №3. – P. 282-294.
9. Mengshoel O.J. Knowledge validation: principles and practice / O.J. Mengshoel, S. Delab // *IEEE Expert*. – 1993. – 8. – P. 62-68.
10. Jahnke J. Genetic fuzzy reasoning nets as a basis for reverse engineering relational database applications [Electronic resource] / J. Jahnke, W. Schafer, A. Zundorf. – Germany, 1998. – 100 p. available at: <http://www.unipaderbom.de/fachbereich/Schaefer/index engl.html>.
11. Hasegama T. Stability analysis of fuzzy control systems based on Petri nets / T. Hasegama, T. Furuhashi, Y. Uchikama // *Proc. Int. Discourse on Fuzzy Logic and the Management of Complexity, FLAMOC'96*. – 1996. – P. 191-195.
12. Nasareth D.L. Investigating the applicability of Petri nets for rule-based system verification / D.L. Nasareth // *IEEE Trans. Software Eng.* – 1993. – 4. – P. 402-415.
13. Polat F. UVT: A unification-based tool for knowledge base verification / F. Polat, H. Guvenir // *IEEE Expert*. – 1993. – 8. – P. 69-75.
14. Koriem S.M. A fuzzy Petri net tool for modeling and verification of knowledge – Based Systems / S.M. Koriem // *The Computer Journal*. – 2000. – 43. – № 3. – P. 206-223.
15. Воевода А.А. Временные сети Петри и диаграммы UML / А.А. Воевода, Д.О. Романников // *Сборник научных трудов НГТУ*. – 2010. – № 1(59). – С. 79-84.
16. Кузьмук В.В. Модифицированные сети Петри и современные методы моделирования параллельных процессов в сложных системах / В.В. Кузьмук, А.В. Кузьмук, О.А. Супруненко, Е.А. Тараненко // *Управління розвитком складних систем*. – 2011. – № 5. – С. 66-72.
17. Коротиков С.В. Применение цветных иерархических сетей Петри для верификации UML-диаграмм на этапе анализа требований к системе дистанционного контроля и управления / С.В. Коротиков // *Сборник научных трудов НГТУ*. – 2007. – № 1(47). – С. 81-92.
18. Марков А.В. Совокупное использование сетей Петри и UML-диаграмм при разработке программного обеспечения / А.В. Марков, Д.О. Романников // *Сборник научных трудов НГТУ*. – 2011. – № 2(64) – С. 85-94.
19. Козюра В.Е. Развертки раскрашенных сетей Петри и их применение для верификации моделей распределенных систем: Автореферат диссертации на соискание ученой степени кандидата физико-математических наук / В.Е. Козюра. – Новосибирск: Институт систем информатики им. А.П. Ершова, 2004. – 19 с.

References

1. Lipp, H.-P. (1991), *Einsatz von zeitbewerteten Fuzzy – Petri-Netzen in Expertensystemen zur operativen Fuehrung komplexer Productionsysteme*, Springer - Verlag, Berlin/Heidelberg/New York, pp. 103-112.
2. Jensen, K. (1986), *Coloured Petri nets, Petri Nets: Central Models and Their Properties*, Springer-Verlag, New York, pp. 248-299.
3. Christensen, S. and Petrucci, L. (1995), *Modular state space analysis of colored Petri Nets, Application and Theory of Petri Nets*, Springer-Verlag, New York, pp. 201-217.
4. Rokyta, P., Fengler, W. and Hummel, T. (1998), *Electronic system design automation using high level Petri nets, Workshop for Hardware Design and Petri Nets*, June 22-26, Lisboa, pp. 129-138.
5. Cao, T. and Sanderson, A. (1993), *Variable reasoning and analysis about uncertainty with fuzzy Petri nets, Application and Theory of Petri Nets*, Springer-Verlag, pp. 126-145.
6. Srinivan, P. and Gracarin, D. (1993), *Approximate reasoning with fuzzy Petri nets, Proc. IEEE INT. Conf. on Fuzzy Systems*, San Francisco, California, USA, pp. 396-401.
7. Scarpelli, H. and Gomide, F. (1993), *Fuzzy reasoning and fuzzy Petri nets, Proc. Fifth IFSA World Congress*, July 5, No. 9, Seoul, Korea, pp. 1326-1329.
8. Scarpelli, H., Gomide, F. and Yager, R.R. (1996), *A reasoning algorithm for high-level fuzzy Petri nets, IEEE Trans. on Fuzzy Systems*, No. 3(4), pp. 282-294.
9. Mengshoel, O.J. and Delab, S. (1993), *Knowledge validation: principles and practice, IEEE Expert*, No. 8, pp. 62-68.
10. Jahnke, J., Schafer, W. and Zundorf, A. (1998), *Genetic fuzzy reasoning nets as a basis for reverse engineering relational database applications*, Germany, 100 p., available at: www.unipaderbom.de/fachbereich/Schaefer/index engl.html.
11. Hasegama, T., Furuhashi, T. and Uchikama, Y. (1996), *Stability analysis of fuzzy control systems based on Petri nets, Proc. Int. Discourse on Fuzzy Logic and the Management of Complexity, FLAMOC'96*, pp. 191-195.
12. Nasareth, D.L. (1993), *Investigating the applicability of Petri nets for rule-based system verification, IEEE Trans. Software Eng.*, No. 4, pp. 402-415.
13. Polat, F. and Guvenir, H. (1993), *UVT: A unification-based tool for knowledge base verification, IEEE Expert*, No. 8, pp. 69-75.
14. Koriem, S.M. (2000), *A fuzzy Petri net tool for modeling and verification of knowledge - Based Systems, The Computer Journal*, No. 3(43), pp. 206-223.
15. Voevoda, A. and Romannikov, D. (2010), “Vremennye seti Petri i diagrammy UML” [Temporary Petri nets and UML diagrams], *Collection of scientific papers NGTU*, No. 1(59), pp. 79-84.

16. Kuz'muk, V., Kuz'muk, A., Suprunenko, O. and Taranenko, E. (2011), "Modifitsirovannyye seti Petri i sovremennyye metody modelirovaniya parallel'nykh protsessov v slozhnykh sistemakh" [Modified Petri nets and modern methods of modeling parallel processes in complex systems], *Managing the development of complex systems*, No. 5, pp. 66-72.

17. Korotikov, S. (2007), "Primeneniye tsvetnykh iyerarkhicheskikh setey Petri dlya verifikatsii UML-diagramm na etape analiza trebovaniy k sisteme dstantsionnogo kontrolya i upravleniya" [Application of color hierarchical Petri nets for verification of UML-diagrams at the stage of analyzing the requirements for the remote monitoring and control system], *Collection of scientific papers NGTU*, No. 1(47), pp. 81-92.

18. Markov, A. and Romannikov, D. (2011), "Sovokupnoye ispol'zovaniye setey Petri i UML-diagramm pri razrabotke programmnoy obespecheniya" [Cumulative use of Petri nets and UML diagrams in software development], *Collection of scientific papers NGTU*, No. 2(64), pp. 85-94.

19. Kozyura, V. (2004), "Razvertki raskrashennykh setey Petri i ikh primeneniye dlya verifikatsii modeley raspredelennykh system: Avtoreferat dySSERTatsyy na soyskaniye uchenoi stepeny kandydata fizyko-matematycheskyykh nauk" [Development of colored Petri nets and their application for verification of models of distributed systems: Abstract of dissertation for the degree of candidate of physical and mathematical sciences], Institute of Informatics Systems named by A.P. Ershov, Novosibirsk, 19 p.

Надійшла до редколегії 4.10.2018

Схвалена до друку 5.11.2018

Відомості про автора:

Тімочко Олександр Олександрович
інженер фірми «Kreditech»,
Гамбург, Німеччина
аспірант Льотної академії Національного
авіаційного університету,
Кропивницький, Україна
<https://orcid.org/0000-0003-0424-0426>

Information about the author:

Oleksander Timochko
Expert Quality Assurance Engineer
of Kreditech Holding,
Hamburg, Germany
Doctoral Student of Flying Academy
of the National Aviation University,
Kropyvnytsky, Ukraine
<https://orcid.org/0000-0003-0424-0426>

ПРЕДСТАВЛЕННЯ ДИНАМІЧНИХ ВЗАЄМОДІЮЧИХ ПРОЦЕСІВ У ПРОСТОРІ СТАНІВ НЕЧІТКОЇ РОЗФАРБОВАНОЇ МЕРЕЖІ ПЕТРІ ДЛЯ ВЕРИФІКАЦІЇ ПРОГРАМНОГО ПРОДУКТУ НЕЧІТКОЇ ЛОГІЧНОЇ СИСТЕМИ КЛАСИФІКАЦІЇ

О.О. Тімочко

Процес верифікації програмного забезпечення класифікації повітряних об'єктів при контролі повітряного простору в нечіткій постановці розглядається як процес дослідження динамічних взаємодіючих процесів. Стверджується, що традиційні підходи не дозволяють із заданою якістю верифікувати програмне забезпечення при рішенні завдань класифікації повітряних об'єктів в автоматизованих системах контролю повітряного простору в заданій постановці. Мета статті – формалізація представлення динамічних взаємодіючих процесів у просторі станів нечіткої розфарбованої мережі Петрі для верифікації програмного забезпечення нечіткої логічної системи класифікації повітряних об'єктів в автоматизованих системах контролю повітряного простору. Проаналізовані основні роботи в області теорії і практики застосування мереж Петрі, використані для дослідження динамічних взаємодіючих процесів загалом, і безпосередньо для верифікації програмного забезпечення різних систем. Висунені вимоги до мереж Петрі для вирішення завдань цього класу. Визначені умови дозволених при представленні взаємодіючих процесів для різних випадків, які відрізняються кількістю вхідних і (чи) вихідних переходів. Для опису компонентів моделі верифікації програмного забезпечення системи класифікації повітряних об'єктів, представлених логікою предикатів, використана нечітка розфарбована мережа Петрі (для логічних операцій I, АБО, НЕ) і відповідні функції приналежності. Розроблений підхід за уявленням динамічних взаємодіючих процесів в просторі станів нечіткою розфарбованою мережею Петрі для верифікації програмного продукту нечіткої логічної системи класифікації є основою для розробки бази нечітких продукційних правил.

Ключові слова: програмне забезпечення, верифікація, класифікація, повітряний об'єкт, взаємодіючий процес, нечітка розфарбована мережа Петрі.

PRESENTING DYNAMIC INTERACTION PROCESSES IN THE SPACE OF THE STATES OF THE FUZZY COLORED PETRI NET FOR VERIFYING THE SOFTWARE OF THE FUZZY LOGICAL CLASSIFICATION SYSTEM

O. Timochko

The process of verifying the software of the classification system for airborne objects in the control of airspace in fuzzy formulation is considered as a process for studying dynamic interacting processes. It is argued that traditional approaches do not allow software to be verified with a given quality when solving the problem of classifying air objects in automated airspace monitoring systems in a given formulation. The purpose of the article is to formalize the presentation of dynamic interacting processes in the state space of a fuzzy colored Petri net for verifying the software of a fuzzy logical system for classifying air objects in automated airspace control systems. The main works in the field of theory and practice of application of Petri nets, used to study the dynamic interacting processes in general, and directly to verify the software of various systems, are analyzed. Requirements for Petri nets have been put forward for solving problems of this class. The conditions for resolving when representing interacting processes for various cases that differ in the number of input and (or) output transitions are determined. A fuzzy colored Petri net (for logical operations AND, OR, NOT) and the corresponding membership functions were used to describe the components of the verification model of the software for the classification of air objects represented by predicate logic. The developed approach to representing dynamic interacting processes in the state space of a fuzzy colored Petri net for verifying a software product of a fuzzy logical classification system is the basis for developing a base of fuzzy production rules.

Keywords: software, verification, classification, air object, interacting process, fuzzy colored Petri network.