

МОДЕЛИРОВАНИЕ ПРИМЕНЕНИЯ ЗНАНИЙ СПЕЦИАЛИСТОВ РАСКРАШЕННЫМИ СЕТЯМИ ПЕТРИ

Димитриев А.П.¹

¹ФГБОУ ВПО «Чувашский государственный университет им И.Н.Ульянова», Чебоксары, Россия (428015, Чебоксары, Московский пр-т, 15), e-mail: dimitrie1@yandex.ru

Задача получения студентами знаний и умений, необходимых на практике, является одной из основных целей функционирования любого вуза. Проблема в том, что со временем эти знания морально устаревают, взамен требуются новые. Данный процесс в работе моделируется раскрашенной временной сетью Петри, учитывающей одновременно нескольких курсов. Достоинство модели – отсутствие ингибиторных дуг. Данная модель является логическим продолжением модели, предложенной автором в предыдущей публикации. При этом она использует предыдущую модель на сетях Петри как основу, представляя с ее помощью иерархические переходы. Каждый такой переход моделирует один курс обучения по своему учебному плану. Используемые атрибуты маркеров новой модели соответствуют атрибутам маркеров предыдущей модели. Рассмотрена связь предложенной модели и моделирования с принципом кэширования в операционных системах. Она наиболее характерна для принципа временной локальности.

Ключевые слова: сеть Петри, атрибут, компетенция, специалист, выпускник, кэш.

MODELING THE APPLICATION OF KNOWLEDGE INHERENT SPECIALISTS USING COLORED PETRI NETS

Dimitriev A.P.¹

¹Federal state budget educational institution of higher professional education "Chuvash State University named after I.N. Ulyanov", Cheboksary, Russia (428015, Cheboksary, Moskovsky Prospect, 15), e-mail: dimitrie1@yandex.ru

The task of obtaining knowledge and skills required in practice is one of the main purposes of the functioning of any institution. The problem is that over time, these skills become obsolete; instead the new ones are required. This process is modeled by colored timed Petri net, simultaneously taking into account several courses. An advantage of the model is the absence of an inhibitor arcs. This model is a logical extension of the model proposed by the author in a previous publication. It uses the previous model on Petri nets as a basis, presenting hierarchical transitions with it assistance. Each such transition simulates one course by its own curriculum. The attributes used tokens of the new model match the attributes of the tokens of the previous model. Considered the relationship of the proposed model and the simulation with the principle of caching in an operating systems.

Keywords: Petri nets, attribute, competence, specialist, graduate, cache.

Как известно, работодатели ожидают от выпускников вуза практических навыков выполнения работ по своей специальности. Для специалистов в информационной сфере деятельности в связи с быстрым развитием информационных технологий эти навыки быстро устаревают. Помочь с этим справиться призвана теоретическая подготовка. Тем не менее, возникает необходимость почти ежегодной корректировки учебных планов в соответствии с потребностями работодателей.

Цель работы – моделирование получаемых студентами теоретических знаний и практических навыков во взаимосвязи с потребностями рынка труда. Для этого применен математический аппарат временных раскрашенных сетей Петри (РСП), наглядно описывающий события в различных дискретных динамических системах, таких как [4], и в том числе при обучении [3].

Разработка математической модели

Смоделируем соотношение качества подготовки специалистов и потребностей рынка рабочей силы помощью РСП (рис. 1). Согласно рабочей документации дисциплины учебного плана формируют свои компетенции, необходимые для профессиональной деятельности (см. стандарт [6]). Номера компетенций обозначим цветом Λ_S маркеров сети Петри. Цвета маркеров соответствуют цветам маркеров из [2] с теми же обозначениями позиций, при этом имеется следующее отличие. Школьные знания смоделированы цветом $X = \{\chi_1, \chi_2, \dots\}$ маркера в позиции P_S , вместо того чтобы использовать несколько позиций $P_{S1}, P_{S2}, \dots, P_{S25}$ и хранить в каждой вещественное число. Тогда в P_S маркер имеет цвет X – вектор вещественных чисел, характеризующих средний уровень знания того или иного школьного предмета. Будем считать, что, так как школьная программа меняется значительно медленнее, чем вузовская, X не меняется. Правила РСП для переходов представлены в табл. 1.

Таблица 1

Правила для переходов

№ п.п.	Переход	Условие	Действие
1.	T_C	Сравнивает Λ_B с Λ_S либо с Ω для α из тех, у которых θ соответствует θ маркера из P_B . Надо, чтобы $\forall \lambda_i \in \Lambda_B$ соответствующий $\chi_j \in X$ у α был $\chi_j \geq x_5$	Срабатывает быстро и означает, что специалист компетент в отношении поступившего поручения и способен его быстро выполнить
2.	T_L	После сравнения T_C , если тот не возбужден	Срабатывает долго. α для соответствующего цвета χ_h увеличивает цвет: $\chi_h = \min\{1, \chi_h + x_3\}$. Если такой компетенции не было, то α , для своего ω_h , наиболее близкого ей по смыслу, добавляет новую часть, и устанавливает цвет $\chi_h = x_4$. Для α , если $\chi_h < x_5$, $\chi_h = \min\{1, \chi_h + x_2\}$. Для всех маркеров в P_K и $\forall h: \chi_h = \chi_h - x_6$.
3.	T_1, T_2, \dots	–	Для T_1, T_2, \dots : для X атрибутов из P_M вычисляется среднее τ . Появляются X маркеров со случайными атрибутами в P_B . G_1 , либо G_2, \dots маркеров помещаются в P_K , приобретая случайным образом Λ_S в зависимости от τ в P_M . Атрибут θ каждого маркера принимает значение атрибута θ одного из маркеров в P_B . Если студенты были на практике на предприятии, при генерации атрибутов Λ_S маркеров в P_K надо дополнять Ω согласно созданному для P_B .

Срабатывание перехода $T_i, i = \overline{1, \dots, n}$ означает, что столько маркеров, сколько студентов в группе (G_i), помещается в позицию P_K , т.е. выпускается группа специалистов. Каждый из этих переходов представляет собой иерархический переход T из [2], но могут быть другие дисциплины и семестры изучения. Чем хуже студент что-то выучит, тем меньше будет цвет χ_i маркера в позиции P_K , где i – номер соответствующей компетенции.

После того, как сработает T_i , в P_B появляются X маркеров. Это означает возможность удовлетворения бывшей группой X профессиональных требований от работодателей. Атрибуты этих маркеров принимают случайные значения, что соответствует случайным повседневным нуждам. Будем считать, что специалисты трудоустраиваются сразу (а незадолго до и не по истечении длительного времени) после выпуска и по специальности. Тогда пусть атрибуты θ соответствующих маркеров в P_K приобретают какие-либо из значений атрибутов θ маркеров в позиции P_B .

Один случайно выбранный маркер в P_K обозначим как α . Он представляет компетенции одного выпускника. В [2] введен ряд числовых констант $x_i, i = \overline{1, \dots, 6}$. Если результаты сравнения (см. п. 1 табл. 1) на удовлетворительном уровне x_5 , то T_C срабатывает быстро. Если запрос не включает ни одну компетенцию, то сравнение производится по области знаний (из некоторого стандартного классификатора).

Рассмотрим п. 2 табл. 1. Если результат сравнения неудовлетворительный, то срабатывает T_L . Это означает обращение к тем или иным внешним источникам информации либо проведение собственных микроисследований. Время срабатывания T_L больше, чем для T_C . При этом освежаются старые знания, в связи с чем в [2] введена константа x_3 , например, $x_3 = 0,2$. Возможно приобретение совершенно нового опыта, для этого в [2] введена константа x_4 , например, $x_4 = 0,5$. Она меньше единицы, потому что для надлежащего запоминания требуется повторение. Знания со временем забываются [7]. Чтобы это учесть, в [2] введена константа забывания, например, $x_6 = 0,0001$. Если знания забыты или устарели, они обновляются с помощью T_L , причем снова не до единицы, а с использованием константы x_2 , например, $x_2 = 0,2$.

Средняя производительность сети Петри оценивает аналогичную величину для реальных специалистов. Чтобы производительность выпускников тотчас после выпуска была на высоком уровне, надо удовлетворить ряд условий, которые требуются для обучения – хорошие расписание, учебный план, материальная база, преподаватели. Например, для составления расписания разработан комплекс программ, использующий наиболее эффективный алгоритм из [1].

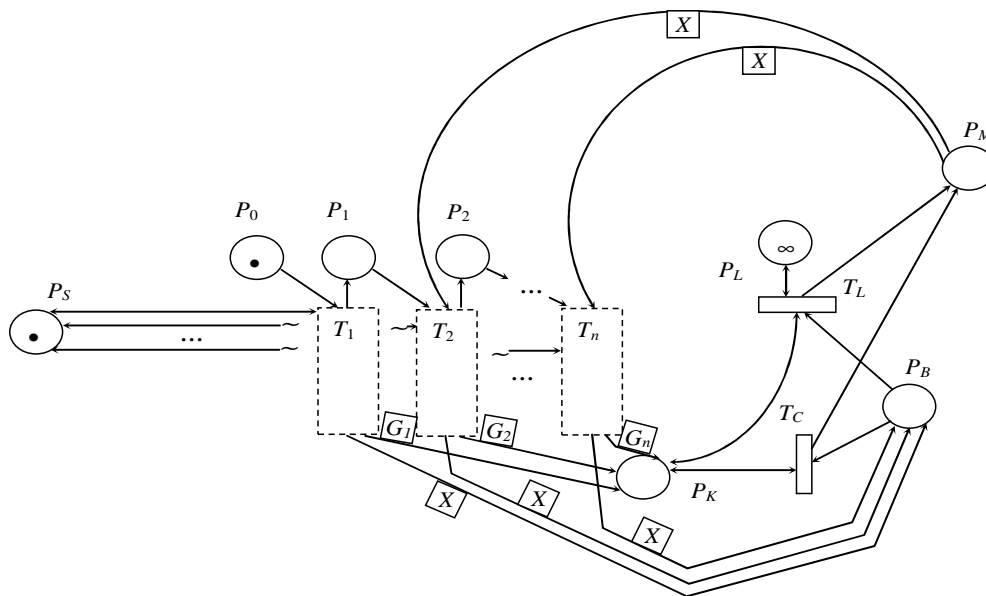


Рис.1. Модель непрерывной подготовки специалистов

Символ \sim означает воображаемой продолжение линии, огибающее препятствие

T_1, \dots, T_n – иерархические переходы (потоки обучения);

X, G_1, \dots, G_n – кратности дуг;

P_M – позиция с информацией о качестве работы выпускников;

T_C – переход, означающий компетентность специалиста;

T_L – переход, означающий обращение в библиотеку или Интернет;

P_B – позиция для хранения профессиональных запросов;

P_K – позиция, моделирующая выпущенных специалистов;

P_S – позиция, моделирующая школьные знания;

P_0, \dots, P_{n-1} – позиции, задающие последовательности приема на 1 курс.

По мере обучения студентов на старших курсах на первый курс поступают новые. У них могут быть другие учебные планы. То есть на факультете одновременно обучаются разные курсы. В сети имеется n переходов T_1, \dots, T_n , каждый из которых соответствует своему потоку обучения (со своим годом приема и учебным планом, по порядку индексов в обозначениях). Последовательность их возбуждения определяется маркировкой позиций P_0, \dots, P_{n-1} . Т.е. исходя из рисунка видно, что первым возбужден T_1 , затем будет T_2 , и т.д.

После возбуждения они, казалось бы, должны работать длительное время, соответственно периоду обучения студентов. Но чтобы не отступать от классических сетей Петри, где срабатывает одновременно только один переход, будем полагать, что это не так. Результат работы сети от этого не меняется. G_1, \dots, G_n – кратности дуг, т.е. количества выпускников.

Одной из целей данного рисунка является показ наличия обратной связи в социальной системе «вуз-работодатели». Позиция P_M означает накопление обществом информации о качестве работы выпускников. Когда очередные X поручений работодателей будут выполнены, там будет X маркеров с различным значением атрибутов τ времени обработки запроса. Потом, используя кратные дуги, все эти атрибуты обрабатываются очередным переходом T_i , т.е. от них зависит Λ_s . τ связано с эффективностью работы специалистов, определяемой работодателями и формирующей то или иное общественное мнение, принимаемое к сведению руководством вуза. Последнее может выплачивать премии преподавателям, делать поручения обновлять оборудование или ремонтировать аудитории в числе факторов, ведущих к успешному обучению.

В связи с тем, что каждый выпуск добавляет разных специалистов с разными компетенциями, в позиции P_K будет все больше и больше маркеров. Выбор среди них для сравнения цветов маркеров производится случайным образом. Большее число специалистов означает, что они могут справиться с большим числом поручений, но если X велико, среднее время обработки сильно не изменится. При генерации значений мест трудоустройства следует ограничивать их общее число небольшим значением, например, 20.

Обычно специалист делает одно и то же, привыкает и делает это быстро. Лишь изредка приходится изучать что-то новое или восстанавливать в памяти забытое. Тут имеется сходство с кэшированием в операционных системах. Известно, что в не менее 90% случаев обращение в компьютере происходит к тем адресам оперативной памяти, содержимое которых продублировано в быстродействующем кэше [5]. Компетенции выпускника соответствуют содержимому кэш-памяти. Что он не знает, или редко использует и забыл, вынужден где-то искать, по аналогии с более медленной основной памятью. основополагающий принцип кэширования - локальность данных во времени – переносится на приведенную модель следующим образом. Если босс недавно поручил что-то сделать, то вероятно, он вскоре поручит делать что-то похожее. Поэтому надо запоминать, что и как делали.

В связи с существованием развитой теории кэширования [5] рассмотрим ее применение для данной модели. Второй основополагающий принцип кэширования – пространственная локальность данных и исполняемого кода, т.е. если было обращение по некоторому адресу, высока вероятность, что скоро потребуется обращение по близлежащему адресу. Значит, если босс что-то требует от подчиненного, возможно, что вскоре он потребует чего-то связанного с первым, но которое выполняется по-другому. Если узкий специалист занимается только своим делом, то делает он его быстро, а хоть и есть дела, связанные с первым, ими должны заниматься другие сотрудники. Самих дел обычно не так много, как адресов в памяти ЭВМ, и в общих чертах можно было бы научиться, как их выполнять, но это затрат-

но, и мало кто на это пойдет. Ради справедливости отметим, что и временные задержки здесь отличаются не на микросекунды, как в ЭВМ, а могут измеряться часами.

Кэширование бывает многоуровневое. Первый уровень соответствует тому, что специалист очень хорошо знает и выполняет, например, еженедельно или ежемесячно. Если нечто требуется выполнять реже, специалист успеваеет подзабыть, но все же он знает, где находится справочная информация и как быстро ее извлечь, это соответствует второму уровню кэша. Новые поручения приходится обдумывать, как выполнить. Специалист может спросить у коллег, но и они могут не знать или из-за недостатка времени не говорить. Тогда к его услугам сеть Интернет, если к ней имеется доступ. Это как бы кэш последнего уровня, после которого необходимы уже радикальные меры для получения информации. Например, можно пойти в библиотеку, или работать методом проб и ошибок, что соответствует обращению к основной памяти ЭВМ.

Существует два основных алгоритма кэширования – сквозной и обратной записи. Имеется в виду, что кэшируется не только чтение, но и запись. Однако, соответственно, большинство специалистов не являются исследователями и новую методику работы обычно не создают, поэтому необходимости в ее запоминании, как правило, нет.

Кэш бывает случайный, когда адреса в нем для кэширования выбираются случайно, и детерминированный, когда эти адреса привязаны к наборам адресов внешней памяти, а также возможна их комбинация. Для специалистов это означает только способ организации ими справочной информации на каких-либо носителях – записных книжках, каталогах файлов и т.п. Например, бывают записные книжки с алфавитными закладками страниц, что эффективно, но со временем они заполняются, теряются, изнашиваются, а также снижается конфиденциальность.

Данные из кэша после его заполнения вытесняются новыми данными, для этого применяются различные алгоритмы замещения страниц, такие как рабочего набора и др. В практике специалистов это применяется в том виде, что время от времени можно выбрасывать устаревшие справочники, старые блокноты, или чистить флэш-накопители. Однако ни о каких алгоритмах для систематизации этих действий не может быть и речи.

Между понятиями «кэширование» и «моделирование» также имеется связь. Моделирование, в том числе математическое, означает работу с моделью, лишенной несущественных свойств объекта моделирования для удешевления эксперимента, например по времени. Кэширование означает сохранение лишь самых актуальных значений из оперативной памяти для ускорения доступа. Это снижает энергопотребление и время, проведенное за компьютером. То есть, как то, так и другое можно направить на облегчение какой-либо работы. Так, представленная на рис. 1 математическая модель абстрагирована от воспитательной и науч-

ной работы, проводимой со студентами. Она не учитывает всех тонкостей учебного плана – его дисциплин, сроков их изучения, форм контроля. Но зато ее значительно легче запрограммировать, чтобы провести вычислительный эксперимент.

Заключение

Достоинством используемой РСР является то, что хотя она и использует известные расширения классических сетей, но не требует применения ингибиторных дуг, а это упрощает логику. Модель является расширением модели [2] для учета обучения на нескольких курсах одновременно. Некоторые из принципов кэширования находят свое применение в повседневной жизни выпускников. Это наиболее характерно для принципа временной локальности, что и нашло отражение в предложенной модели. Для модели кэширования [3, С. 55] проведен вычислительный эксперимент, показывающий соответствие результатов ожидаемым числовым значениям согласно [5].

Список литературы

1. Димитриев А.П. Модели и алгоритмы в системах автоматизированного перевода текста // Прикладная информатика. – 2013. - № 6 (48). – С. 45-59.
2. Димитриев А.П. Моделирование усвоения и применения учебного материала на раскрашенных сетях Петри // Современные проблемы науки и образования. – 2014. - № 4. URL: <http://www.science-education.ru/118-14367> (дата обращения: 20.08.2014).
3. Желтов В.П., Димитриев А.П. Стохастическая оптимизация расписания на сетях Петри. – Чебоксары: Изд-во Чуваш. ун-та, 2001. – 213 с.
4. Желтов П.В. Модели поиска и копирования символьных данных на J-сетях // Прикладная информатика. – 2012. - №4 (40). – С. 81-83.
5. Олифер В.Г., Олифер Н.А. Сетевые операционные системы: [учебник для вузов]. – СПб. и др.: Питер, 2009. – 669 с.
6. ФГОС ВПО по направлению подготовки 230700 «Прикладная информатика». – М.: Минобрнауки РФ, 2009. URL: http://www.edu.ru/db-mon/mo/Data/d_09/prm783-1.pdf (дата обращения: 28.05.2014).
7. Ebbinghaus H. Memory. A Contribution to Experimental Psychology. New York: Teachers College, Columbia University, 1913. (Reprinted Bristol: Thoemmes Press, 1999). URL: <http://psy.ed.asu.edu/~classics/Ebbinghaus/index.htm> (дата обращения: 11.06.2014).

Рецензенты:

Белов В.В., д.т.н., профессор, декан инженерного факультета ФГБОУ ВПО «Чувашская государственная сельскохозяйственная академия», г. Чебоксары;

Захаров В.М., д.т.н., профессор кафедры компьютерных систем Казанского национального исследовательского технического университета им. А.Н. Туполева (КАИ), г. Казань.