

*Е.А. Козлова, А.А. Сукощников
Вологодский государственный университет*

РАЗРАБОТКА СИСТЕМЫ МОДЕЛИРОВАНИЯ НА БАЗЕ НЕЧЕТКО-НЕЙРОННЫХ СЕТЕЙ ПЕТРИ

В статье рассматриваются вопросы проектирования системы моделирования на базе нечетко-нейронных сетей. При этом система должна позволить исследовать обычные, временные, стохастические, нечеткие, нейронные сети и учитывать использование ингибиторных дуг. Для разработки применяется компонентно-ориентированный подход движка Unity.

Сеть Петри, нечетко-нейронная сеть, движок Unity.

К одним из инструментов имитационного моделирования относят сети Петри, которые впервые были предложены Карлом Адамом Петри в 1962 году в его докторской диссертации «Связь с автоматами».

Сеть Петри – это двудольный ориентированный мультиграф, состоящий из вершин двух типов – позиций и переходов, соединенных между собой дугами [1–5]. При этом вершины одного типа («позиция – позиция» или «переход – переход») не могут непосредственно соединяться между собой дугами. В позициях могут размещаться метки (маркеры), которые могут перемещаться по сети. Положение меток в сети описывает состояние сети в тот или иной момент модельного времени.

Компонентно-ориентированный подход движка Unity позволяет обособиться от проектирования среды целиком, разделить ее на независимые элементы со своей логикой работы. Этот подход очень близок сетям Петри, где каждый элемент отвечает только за свое состояние [1, 2].

Укажем преимущества и недостатки движка применительно к разрабатываемому программному продукту:

- **Преимущества:**

1. Существование префабов значительно упрощает работу над сложными элементами и связями внутри них, предоставляя возможности компонентно-ориентированного программирования.

2. Движок работает под управлением более чем двадцати операционных систем.

3. Доступность движка позволяет использовать бесплатно практически все его возможности, предоставляет обширную справочную информацию и развитое русскоязычное комьюнити.

- **Недостатки:**

1. Фиктивная независимость элементов сети Петри друг от друга требует создания большого количества ссылок для координатора работы сети и для функций изменения и вывода характеристик.

2. При больших сценах и большом количестве элементов резко снижается производительность, за-

ставляя тратить время на оптимизацию, в том числе и размера проекта.

- **Другие особенности:**

1. Большое количество шаблонов и библиотек предоставляют большой встроенный функционал, но вместе с тем требуют внимательной проверки, какие именно из них на самом деле нужны в проекте.

2. Объединение 2D и 3D часто приводит к появлению артефактов и неочевидному обращению к размерам и положению объектов.

Движок Unity поддерживает язык программирования C#, предоставляющий все возможности объектно-ориентированного подхода. Такие возможности данного языка программирования, как динамическая очистка мусора, наличие ссылочных данных, наследование (при отсутствии множественного наследования), давно завоевали расположение программистов. В наши дни язык программирования C# является одним из самых популярных языков программирования, по данным поиска Google входя в ТОП-5 после JavaScript, Python, Java и TypeScript (первые два языка из списка используются для программирования интерактивных страниц в браузере и анализа данных соответственно, поэтому не подходят для наших целей).

Разработка алгоритма работы среды моделирования

Логика сетей Петри является самой важной частью для разрабатываемой среды моделирования, ведь именно она отвечает за непосредственно моделирование, что является нашей целью. Всю логику сетей можно разделить на четыре блока:

1. Непосредственно сами сети Петри (назовем их обычными сетями Петри), представляющие собой временную и стохастическую составляющие. Они будут основой среды, на которой будут строиться остальные составляющие.

2. Нечеткая составляющая сетей Петри.

3. Нейронная составляющая сетей Петри.

4. Нечетко-нейронная составляющая, объединяющая в себе логику нечетких и нейронных сетей [1–5].

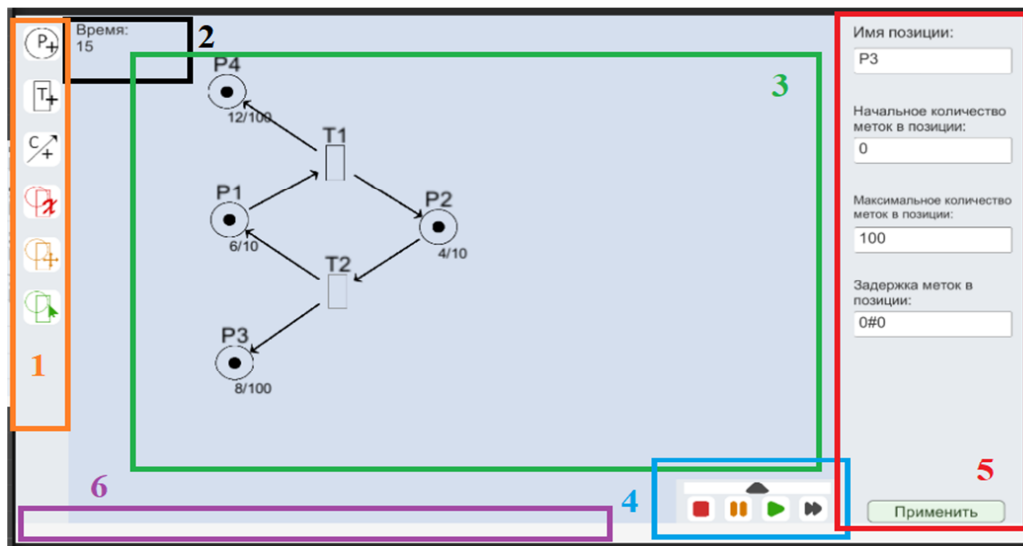


Рис. 1. Редактор разрабатываемой программы моделирования сетей Петри

Нечеткой, нейронной и нечетко-нейронной составляющим в рамках этой научно-исследовательской работы дадим возможность создания ингибиторных дуг. Помимо этого, опишем, какие переменные и связи будут необходимы для каждого элемента каждой составляющей сети.

Для обеспечения независимой работы составляющих различных сетей Петри следует создать отдельные панели свойств: панели для элементов, соответствующих всем разновидностям СП (универсальные панели свойств), панели для обычной СП, панели для нечеткой СП, панели для нейронной СП, панели для нейро-нечеткой СП.

В универсальную панель свойств позиции добавим два чекбокса, позволяющих переключаться между различными сетями Петри и, соответственно, между панелями свойств. Сами переменные, соответствующие этим чекбоксам, будут храниться в скрипте-координаторе, там же изменяться и оттуда же считываться.

Разработка пользовательского интерфейса

Перейдем к проектированию более удобного в использовании пользовательского интерфейса, который показан на рисунке 1. Опишем то, что мы видим, и то, как это планируется реализовать.

Разработка нечеткой составляющей сети Петри

Для разрабатываемой среды моделирования воспользуемся НСП типа C_f .

Изменения, которые коснутся позиций. Каждая позиция может определяться десятичной дробью со значением от 0 до 1 включительно, что соответствует значению функции принадлежности (ЗФП) нечеткого наличия одного маркера в ней. Таким образом, в панель нечеткой позиции следует добавить лишь одно поле, а в ее скрипт вписать различные поведения в зависимости от значения переменной, отвечающей за нечеткость, скрипта-координатора [3].

Изменения, которые коснутся переходов. В панель свойств нечеткого перехода следует добавить два поля для ввода десятичных дробей: мера возмож-

ности нечеткого срабатывания (МВНС) и порог срабатывания перехода. Переход может сработать при условии, что во всех его входных позициях ЗФП нечеткого наличия одного маркера в позиции не меньше порога срабатывания перехода. Мера возможности нечеткого срабатывания перехода потребуется для скрипта изменения маркировки выходных позиций сработавшего перехода.

При срабатывании нечеткого перехода состояние нечеткой позиции будет определяться одним из трех способов:

1. Для входных позиций перехода значение функции принадлежности приравнивается нулю.

2. Для выходных позиций перехода новая маркировка определяется по следующим правилам:

А. Выбирается минимальное значение функций принадлежности входных позиций перехода (легче всего добавить эту функцию в скрипт перехода).

Б. Выбранное значение сравнивается с мерой возможности нечеткого срабатывания перехода (эта функция тоже будет реализована в скрипте перехода). Выбирается наименьшее из этих чисел.

В. Полученное число сравнивается со значением функции принадлежности выходной позиции перехода и из них выбирается наибольшее, которое и становится новым значением функции принадлежности выходной позиции. Эта функция срабатывает независимо для каждой из выходных позиций, поэтому ее лучше добавить в скрипт позиций.

3. Для позиций, не являющихся для данного перехода ни входными, ни выходными, состояние не изменяется.

Изменения, которые коснутся дуг. Так как маркировка сети может интерпретироваться как вектор вероятностей наличия одного маркера в соответствующей позиции, дуги не могут иметь кратность, отличную от единицы, что следует описать в их скрипте.

Следует помнить, что в нечеткой сети Петри типа C_f нет временной составляющей.

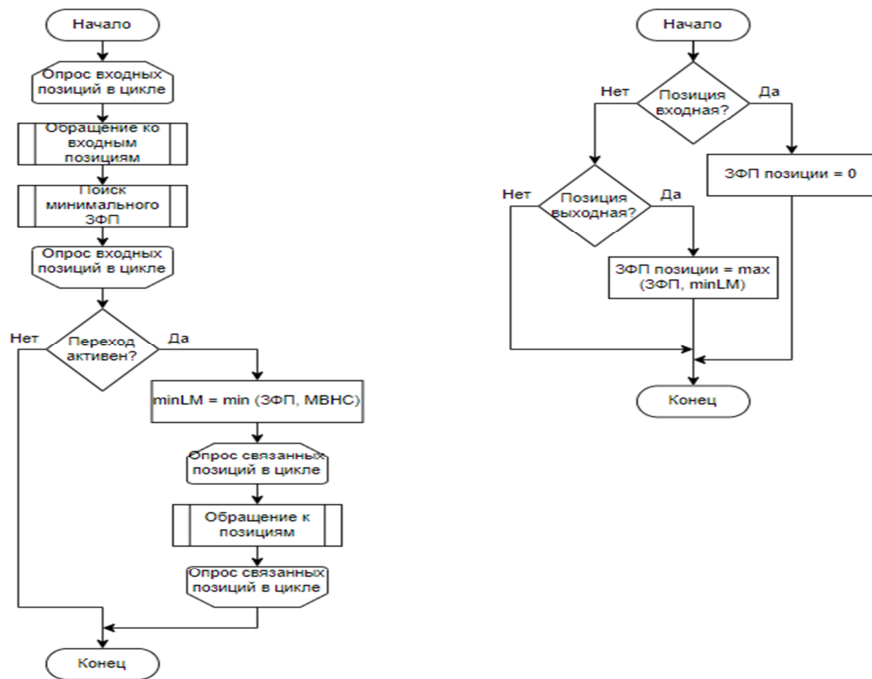


Рис. 2. Блок-схемы алгоритмов работы нечеткого перехода (слева) и нечеткой позиции (справа)

Разработка нейронной составляющей сетей Петри

Опишем логику работы нейронной сети применительно к разрабатываемой среде моделирования и укажем дополнительные параметры и функции, которые стоит добавить к уже имеющимся после предыдущих пунктов разработкам.

Во-первых, в универсальную панель свойств позиции следует добавить чекбокс, позволяющий включать и выключать нейронную составляющую в сети.

Во-вторых, каждой метке должен быть присвоен свой постсинаптический потенциал. Этот потенциал может быть изменен при достижении меткой позиции. Таким образом, в панель свойств нейронных позиций следует добавить поле, отвечающее за прибавление ПСП меток, достигших данных позиций. Для меток, находящихся в такой позиции при старте моделирования, начальный ПСП примет указанное в этом поле значение.

Помимо этого, нейронные позиции вместе с максимальным количеством меток обладают свойством «количество меток для активизации перехода». Суммарный ПСП в таких позициях считается только при достижении указанного количества меток, и переход, для которого такая позиция является входной (нейронный переход) срабатывает на основании этого суммарного ПСП. Для достижения данного результата следует в панель свойств нейронных позиций добавить поле «количество меток для активизации перехода», а в скрипт самой позиции добавить функции, подсчитывающие ПСП по формуле, аналогичной (1) (формула для четырех меток в позиции):

$$f(P) = \frac{4}{4}\mu(P_i) + \frac{3}{4}\mu(P_{i-1}) + \frac{2}{4}\mu(P_{i-2}) + \frac{1}{4}\mu(P_{i-3}). \quad (1)$$

Здесь $\mu(P_i)$ – ПСП метки, прибывшей в позицию последней, $\mu(P_{i-3})$ – ПСП метки, прибывшей в позицию первой. Если метка прибыла в позицию с отрицательным потенциалом, это также учитывается.

В-третьих, нейронные переходы обладают порогом срабатывания. Его можно обозначить как условие: переход срабатывает, когда суммарный ПСП меток нейронной позиции соответствует условию «больше, меньше или равно определенному значению». Для этого стоит добавить поле ввода порога срабатывания в панель изменения свойств нейронного перехода и изменить скрипт определения возможности срабатывания перехода.

В-четвертых, дуги могут изменять цвет меток. В нашем случае это будет менять знак ПСП: ПСП будет либо положительным, либо отрицательным. Для этого следует добавить в панель свойств нейронных дуг поле изменения цвета метки и соответствующую функцию в скрипт самой дуги. Следует помнить, что дуги теперь будут передавать не только количество меток, но и массив их ПСП [4, 5].

Дополнительно следует задуматься над случаями, когда из нескольких возможных меток вперед перемещаются лишь некоторые из них. Возможно, стоит сортировать метки по времени их прибытия и позволить пользователю выбрать правило FIFO (метки, пришедшие раньше, первыми покинут позицию) либо LIFO (метки, пришедшие последними, первыми покинут позицию).

Разработка нейро-нечеткой составляющей сетей Петри

Нечетко-нейронные сети Петри не являются распространенным явлением в моделировании, поэтому придется самостоятельно вывести правила работы таких сетей.

Нечеткость в нейронной сети будет задаваться нечетким ПСП меток (нечеткий ПСП находится в диапазоне [0, 1]).

Изменения, которые коснутся позиций. Позиции все еще хранят время, когда каждая метка будет обработана. О метках она также будет знать и массив их ПСП. Помимо этого, от нейронных сетей остается

свойство «количество меток для активизации перехода». Аналогично нейронным позициям, нейронная нейро-нечеткая позиция высчитывает суммарный ПСП, когда количество меток в ней больше или равно значению переменной «количество меток для активизации перехода». Формула для расчета суммарного ПСП была показана выше (1).

При срабатывании нейронного нейро-нечеткого перехода, для которого нейронная позиция является входной, позиция опустошается. В выходные позиции того же перехода помещается количество меток, равное кратности выходящей дуги, чей ПСП равен суммарному ПСП нейронного перехода.

При срабатывании перехода, не связанного с нейронной нейро-нечеткой позицией, входная ненейронная нейро-нечеткая позиция теряет количество меток, соответствующее кратности входящей в переход дуги. Выходная же ненейронная нейро-нечеткая позиция получает массив ПСП, чье количество элементов соответствует кратности выходящей из перехода дуги, добавляет каждому элементу ПСП по формуле (2) и прибавляет получившийся массив к концу собственного массива ПСП:

$$\mu(m_i) = \max(\mu(m_i), \mu(P_i)). \quad (2)$$

В формуле (2) $\mu(m_i)$ – ПСП метки, а $\mu(P_i)$ – добавочный ПСП позиции. Другими словами, метки, чей ПСП меньше добавочного ПСП позиции, получают ПСП, равный добавочному ПСП позиции.

Таким образом, нейро-нечеткие позиции обладают следующими свойствами: начальное количество маркеров в позиции, максимальное количество маркеров в позиции, количество меток для активизации перехода, время обработки маркера в позиции и ПСП меток, входящих в позицию.

Изменения, которые коснутся переходов. Имя и приоритет перехода являются обязательными полями для переходов всех типов сетей. Для нечетких переходов характерны поля «мера возможности нечеткого срабатывания» и «порог срабатывания перехода». В случае нейронных нейро-нечетких переходов высчитанный в нейронной позиции суммарный ПСП сравнивается с условием срабатывания перехода аналогично нейронным переходам. Соответствие суммарного ПСП позиции условию срабатывания перехода позволяет переходу сработать. Сработавший нейронный переход сравнивает ПСП каждой метки, проходящей через него со своей мерой возможности нечеткого срабатывания, и, если ПСП метки выше, он приравнивается к МВНС.

Ненейронный нейро-нечеткий переход при срабатывании пропускает через себя только те метки, чей ПСП выше или равен его порогу срабатывания. Аналогично нейронным нейро-нечетким переходам, метки, чей ПСП выше МВНС перехода, получают ПСП, равный МВНС перехода.

Изменения, которые коснутся дуг. Аналогично нейронным сетям, дуги в нечетко-нейронных сетях обладают кратностью и цветом. Нечетко-нейронные дуги также передают в позиции и переходы массивы ПСП меток, изменяя знак ПСП в соответствии со своим цветом.

Таким образом, разработана система, позволяющая моделировать временные, стохастические, нечеткие, нейронные и нечетко-нейронные сети Петри. В правила работы всех типов сетей была добавлена возможность создавать ингибиторные дуги, что для нечетких и нейронных сетей является достаточно новым и неожиданным шагом. Анимация моделирования даст возможность наглядно увидеть передвижение меток по сети, что поможет новичкам на первых порах изучения сетей Петри.

Литература

1. Козлова, Е. А. Особенности движка Unity при создании среды моделирования сетей Петри / Е. А. Козлова // ИНФОС-2023. – Вологда, 2023. – С. 103–113.
2. Козлова, Е. А. Особенности движка Unity при создании среды моделирования сетей Петри / Е. А. Козлова // Школа практических инноваций – инженерному бизнесу региона : материалы Всероссийской научно-практической конференции памяти профессора Александра Николаевича Шичкова – Вологда, 2024. – С. 160–164.
3. Суконщиков, А. А., Кочкин, Д. В. Нечеткий вывод на базе аппарата модифицированных нечетких сетей Петри / Вестник Череповецкого государственного университета. – Череповец : ЧГУ, 2013. – Т. 2. – № 3 (50). – С. 24–28.
4. Суконщиков, А. А. Разработка и применение нейро-нечетких сетей Петри / А. А. Суконщиков, А. Н. Швецов, А. В. Улитин // Вестник Череповецкого государственного университета. – 2020. – № 2. – С. 35–45.
5. Суконщиков, А. А., Кочкин, Д. В. Методика создания моделей на базе модифицированных сетей Петри / Программные продукты и системы – Научное и научно-практическое издание. – Тверь: НИИ «Центрпрограммсистем», 2013. – № 3. – С. 248–251.

*E.A. Kozlova, A.A. Sukonshchikov
Vologda State University*

DEVELOPMENT OF MODELING SYSTEM BASED ON FUZZY NEURAL PETRI NETWORKS

The article discusses issues of designing a modeling system based on fuzzy neural networks. At the same time, the system should allow the study of ordinary, temporary, stochastic, fuzzy, neural networks and take into account the use of inhibitory arcs. The component-oriented approach of the Unity engine is used for development.

Petri net, fuzzy neural network, Unity engine.