**Программный комплекс конструирования паттернов поведения роботов**

Разработка алгоритмов поведения робота является сложной задачей, решение которой должно учитывать множество факторов: данные об окружающей среде, технические характеристики аппаратной платформы и глубокое понимание задачи, стоящей перед роботом. Как правило, все особенности способен учесть только профессионал ˗ эксперт в конкретной предметной области. Однако, далеко не всегда он способен запрограммировать робота, этим должен заниматься программист [1].

На сегодняшний день проектирование и разработка любого программного обеспечения, в том числе программирование робота, состоит как минимум из двух этапов: проектирование и программирование. Проектирование осуществляется посредством графического описания будущей системы в виде диаграмм, представленных в той или иной нотации. Программирование робота осуществляется на основе спроектированных графических моделей с использованием различных языков программирования. Как правило, алгоритм работы описывают с помощью блок-схемы. Однако нарисованная диаграмма может иметь множество ошибок и недостатков, приводить к конфликтам в системе. В связи с этим, пользователю необходимо предоставить возможность программирования робота с помощью простейших операций, не требующих специальных навыков. В том числе, требуется разработать инструмент, объединяющий этапы проектирования и программирования.

Целесообразно создать такую среду, в которой каждый добавленный пользователем графический примитив будет знать о своем предназначении и связях с другими сущностями. Это позволит применить к диаграмме какой-либо математический аппарат, например сети Петри, модели которых позволяют исследовать работоспособность моделируемых систем, оптимальность их структуры, эффективность процесса их функционирования, а также возможность достижения в процессе функционирования определенных состояний, что в свою очередь позволит сделать выводы о сконструированном экспертом алгоритме поведения робота.

Сети Петри (СП) и их многочисленные модификации являются одним из классов моделей, неоспоримым достоинством которых является возможность адекватного представления не только структуры сложных организационно-технологических систем и комплексов, но также и логико-временных особенностей процессов и их функционирования, что особенно актуально при моделировании поведения робота. Сети Петри представляют собой математическую модель для представления структуры и анализа динамики функционирования систем в терминах «условие-событие». Эта модель может быть успешно использована для описания так называемых динамических дискретных систем различных классов, таких как: вычислительные процессы и программы, технологические процессы, информационные, экономические, биологические, социальные и технические системы.

Сети Петри и их обобщения являются удобным и мощным средством моделирования асинхронных, параллельных распределительных и недетерминированных процессов, позволяют наглядно представить динамику функционирования систем и составляющих их элементов. Свойство иерархического вложения сетей Петри позволяет рассматривать модели различной степени детализации, обеспечивая тем самым необходимую декомпозицию сложных систем и процессов [5].

При моделировании процессов функционирования дискретных динамических систем, исследование характеристик сетей Петри предполагает решение следующих задач [6]:

1. Задача достижимости маркировок: для заданной СП C = (P,T,I,O,$ m\_{0}$) и заданной маркировки $m\_{r}$ установить выполнение условия $m\_{r}\in D(m\_{0})$.
2. Задача достижимости подмаркировки: для заданной СП C = (P,T,I,O,$ m\_{0}$) и заданной маркировки $m\_{r}$ и подмножества $P^{'}\in P$ определить существует ли достижимая маркировка $m\_{w}$, компоненты вектора которой с номерами позиций из подмножества $P^{'}$ равны соответствующим компонентам вектора маркировки $m\_{r}$. Остальные компоненты вектора $m\_{r}$ могут принимать произвольные значения.
3. Задача покрываемости маркировки: для заданной СП C = (P,T,I,O,$ m\_{0}$) и заданной маркировки $m\_{r}$ определить, существует ли достижимая маркировка $m\_{w}\in D(m\_{0})$, такая что $m\_{w}\geq m\_{r}$.
4. Задача достижимости нуля: для заданной СП C = (P,T,I,O,$ m\_{0}$) определить, достижима ли нулевая маркировка $m$ = (0,0,…,0) из начальной маркировки $m\_{0}$.
5. Задача достижимости нуля в одной позиции: для заданной СП C = (P,T,I,O,$ m\_{0}$) и некоторой фиксированной позиции $p\_{i}\in P$ определить, существует ли достижимая маркировка $m\in D(m\_{0})$, в которой $m\_{i}=0$.
6. Задача равенства: для двух СП C’ = (P’,T’,I’,O’,$ m\_{0}$’) и C’’ = (P’’,T’’,I’’,O’’,$ m\_{0}$’’), таких что $P^{'}= P^{'}'$, определить, равны ли соответствующие им множества достижимых маркировок $D\left(m\_{0}^{'}\right)= D(m\_{0}'')$.
7. Задача подмножества: для двух СП C’ = (P’,T’,I’,O’,$ m\_{0}$’) и C’’ = (P’’,T’’,I’’,O’’,$ m\_{0}$’’) определить, является ли одно из множеств достижимых маркировок подмножеством другого.
8. Задача K-ограниченности позиций и СП в целом: для заданной СП C=(P,T,I,O,$ m\_{0}$) и фиксированной позиции $p\_{i}\in P$ определить, существует ли натуральное число К, для которого позиция $p\_{i}$ является К-ограниченной.
9. Задача безопасности позиций и СП в целом: для заданной СП C = (P,T,I,O,$ m\_{0}$) и фиксированной позиции $p\_{i}\in P$ определить, является ли позиция $p\_{i}$ безопасной.
10. Задача устойчивости переходов: для заданной СП C = (P,T,I,O,$ m\_{0}$) и заданного подмножества переходов T’, определить являются ли переходы из подмножества устойчивыми.
11. Задача активности переходов: для заданной СП C = (P,T,I,O,$ m\_{0}$) и заданного подмножества переходов T’ определить уровни активности переходов заданного подмножества.
12. Задача сохраняемости СП: для заданной СП C = (P,T,I,O,$ m\_{0}$) определить, является ли она сохраняющей.
13. Задача строгой сохраняемости СП: для заданной СП C = (P,T,I,O,$ m\_{0}$) определить, является ли она строго сохраняющей.

Сконструированный алгоритм работы интеллектуального агента с последующим преобразованием его в сеть Петри, с учетом выше перечисленных задач, позволяет получить сведения о следующих характеристиках:

1. о количестве процессов в системе;

2. наличии взаимоблокировок;

3. наличии невыполнимых операций;

4. количестве циклов, которые при определенных ситуациях могут стать причиной зацикливания.

Разрабатываемая система предполагает конструирование алгоритмов поведения роботов путем рисования блок-схем с последующей трансляцией их в сети Петри. Блок-схема во многом подобна сети Петри. В первую очередь она представима в виде узлов двух типов (принятия решения, показанные ромбами, и вычисления, показанные прямоугольниками) и дуг между ними. Удобный способ выполнения блок-схемы ˗ введение фишки, которая представляет текущую инструкцию. По мере выполнения инструкций фишка передвигается по блок-схеме. Перевод блок-схемы в сеть Петри заменяет узлы блок-схемы на переходы сети Петри, а дуги блок-схемы ˗ на позиции сети Петри. Каждая дуга блок схемы соответствует точно одной позиции в сети Петри. Узлы блок-схемы представляются по-разному в зависимости от типа узла: вычисления или принятия решения. Фишка, находящаяся в позиции, означает, что счетчик команд установлен на готовность выполнения следующей инструкции. Каждая позиция имеет единственный выходной переход, за исключением позиции, которая имеет по два выходных перехода, соответствующих истинному и ложному значению предиката [2]. Для интерпретации сети Петри необходимо отображать каждый переход. Следует также отметить, что переходы для вычислений имеют по одному входу и выходу. В таблице 1 иллюстрируются оба способа перевода [3,4].

Таблица 1 ˗ Перевод узлов вычисления и принятия решения в блок-схеме в переходы сети Петри

|  |  |
| --- | --- |
| Элемент блок-схемы | Вид композиции в виде сети Петри |
|  |  |
|  |  |

На рисунке 1 показ пример блок-схемы включения робота и его аналог в виде сети Петри. Данный пример может быть использован практически в любом описании алгоритма робота и выступать в качестве подготовительного этапа.

…

Рисунок 1 ˗

Таким образом, создание системы генерации алгоритмов поведения робота с использованием механизма конструирования с предварительным преобразованием блок-схемы в Сеть Петри позволит ускорить разработку прошивок и их качество при решении задач различного назначения.

1. Сорокин Е.В. Визуального среда конструирования алгоритмов поведения интеллектуального агента

2. Котов В.Е. Сети Петри. М.:Наука, Гл. ред. физ.-мат. лит., 1984. 160с.

3. Рудаков И.В., Пащенкова А.В. Программный комплекс верификации алгоритмов программного обеспечения с помощью иерархических сетей Петри. СГТУ им. Н.Э. Баумана, Москва 105005, Россия. 10с.

4. Норенков И.П. Основы автоматизированного проектирования: учеб. для вузов. М.: Изд-во МГТУ им. Н.Э. Баумана, 2002. 306 с.

5. Питерсон Дж. Теория сетей Петри и моделирование систем: пер. с англ. М.: Мир. 1984. 264с.

6. Леоненков А.В. Нечеткое моделирование в среде MATLAB и fuzzyTECH. – СПб.: БХВ-Петербург, 2005. – 736 с.