

№ 4. – С. 3-17.

5. Поздняков, А. Д. Анализ возможностей вычислительного метода определения девиации частоты сигнала на основе преобразований Гильберта и Фурье / А. Д. Поздняков, В. А. Поздняков, Д. С. Коробов // Известия Института инженерной физики. – 2010. – № 1(15). – С. 33-37.

6. Wiener N. Hermit polynomials and Fourier analysis // Journal of Mathematical Physics MIT. 1929. V. 8. P. 70–73.

7. Almeida L.B. The fractional Fourier transform and time-frequency representations // IEEE Transactions on Signal Processing. 1994. V. 42. P. 3084–3091.

УДК 681.324:621.391.1

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ ПРОЦЕССОВ В САПР НА ОСНОВЕ PRO-СЕТЕЙ

Горячев Александр Владимирович

канд. техн. наук, доц., доцент кафедры систем автоматизированного проектирования

Новакова Наталья Евгеньевна

канд. техн. наук, доц., доцент кафедры систем автоматизированного проектирования

ФГАОУ ВО «Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)»

Аннотация. Рассмотрены особенности процесса автоматизированного проектирования. Представлена модель процессов в САПР на основе модификации сетей Петри. Описана дискретно-событийная модель типовой проектной процедуры.

Ключевые слова: САПР; модифицированные сети Петри; имитационное моделирование; позиции; переходы; ресурсы.

Abstract. The features of the computer-aided design process are considered. A model of processes in CAD based on the modification of Petri nets is presented. A discrete event model of a typical design procedure is described.

Keywords: CAD; modified petri nets; simulation; positions; transitions; resources.

Введение. Процесс создания и внедрения САПР является трудоемкой и длительной задачей, требующей участия коллектива высококвалифицированных специалистов: программистов, экспертов, консультантов, инженеров знаний и их согласованных действий с коллективом проектировщиков и администрацией проектной организации. Особо следует выделить начальный этап разработки средств автоматизации проектирования и выбора ресурсов, имеющий большое значение для получения требуемого эффекта от внедрения САПР в конкретных условиях эксплуатации. Ошибочное, недостаточно продуманное решение по выбору типа компонентов системы и способов их взаимодействия, принятое на ранних стадиях разработки системы, может привести к непоправимым последствиям: существенным изменениям подсистем, замене компонентов, а возможно, и к полной реконфигурации всей системы. Следовательно, необходимо применение специальных инструментальных средств автоматизации разработки архитектуры САПР и оценки качества проектных решений на ранних этапах разработки. Такие инструментальные средства полезны как на ранних этапах проекта, так и при модификации существующей системы. В качестве рабочих моделей исследования конфигурации САПР можно использовать: модели, построенные на графах; логику предикатов, алгоритмические модели; модели, использующие сети Петри. Наиболее подходящей теоретической основой имитационной модели процесса автоматизированного проектирования является применение

модификацию сетей Петри, позволяющую учитывать временные характеристики процесса.

Целью данной статьи является разработка имитационной модели процесса автоматизированного проектирования с применением модифицированных сетей Петри.

Изложение основного материала.

Процесс автоматизированного проектирования должен обладать следующими свойствами: структурированность; итерационность; целенаправленность; стохастичность; длительность [1]. Помимо вышеперечисленных свойств процесс проектирования должен быть дискретным, динамичным и иметь возможность отражать параллельное выполнение проектных процедур.

Дискретно-событийное моделирование используется для дискретных во времени процессов низкой и среднего уровня абстракции. Дискретно-событийную модель можно рассматривать как схему обслуживания заявок. Необходимо разработать логико-математическую модель, которая позволяет имитировать выполнение проектных процедур [1]. Дискретно-событийная модель позволяет организовать последовательное выполнение типовых проектных процедур в том же порядке, в котором они выполнялись бы в реальном процессе автоматизированного проектирования. При моделировании основным объектом в системе является токен (имитация заявки), который может определенным образом «захватывать» ресурсы, необходимые для выполнения типовой проектной процедуры и имеет определенные характеристики. «Перемещаясь» по блоку, токены становятся в очередь на выполнение к блоку, захватывают и освобождают ресурсы.

В имитационной модели обрабатывается токен – проектная задача. У каждого токена есть следующие характеристики: сложность объекта проектирования; прогресс выполнения процедуры; время начала обработки данных блоком; время окончания процедуры.

Разница между временем начала и окончания представляет собой длительность выполнения задачи. В общем виде алгоритм можно описать с помощью нескольких шагов:

Шаг 1: Запуск стартового токена.

Шаг 2: Токен поступает в очередь обработки первого блока.

Шаг 3: Если прогресс выполнения, обновляющийся итерационно через d_i , становится равным 1 (100%), то переход на шаг 4.

Шаг 4: Токен удаляется.

Шаг 5: На выходе из первого блока появляется новый токен.

Таким образом, в конце каждой итерации процесс перемещает токены с выходов блоков на входы связанных с ними блоков (рис. 1). В модели есть начальный и конечный блок – начальная и конечная проектная процедура соответственно. У подобных блоков есть особенность: начальный блок не имеет входов, конечный – выходов. При моделировании происходит изменение модельного времени, отличного от реального. Модельное время дискретно, измеряется в тактах. Изменение времени происходит по окончании имитации блока, относящегося к текущему моменту времени. Моделирование процесса проектирования заканчивается, когда исчерпано общее время или выполнены все процедуры.

Сети Петри – это аппарат для моделирования динамических систем и асинхронных параллельных процессов [2]. В сетях Петри используются объекты двух типов: статические и динамические.

Основные объекты, посредством которых выстраивается сеть:

- маркеры (динамические метки внутри вершин сети Петри);
- вершины (статический элемент);
- дуги (статический элемент).

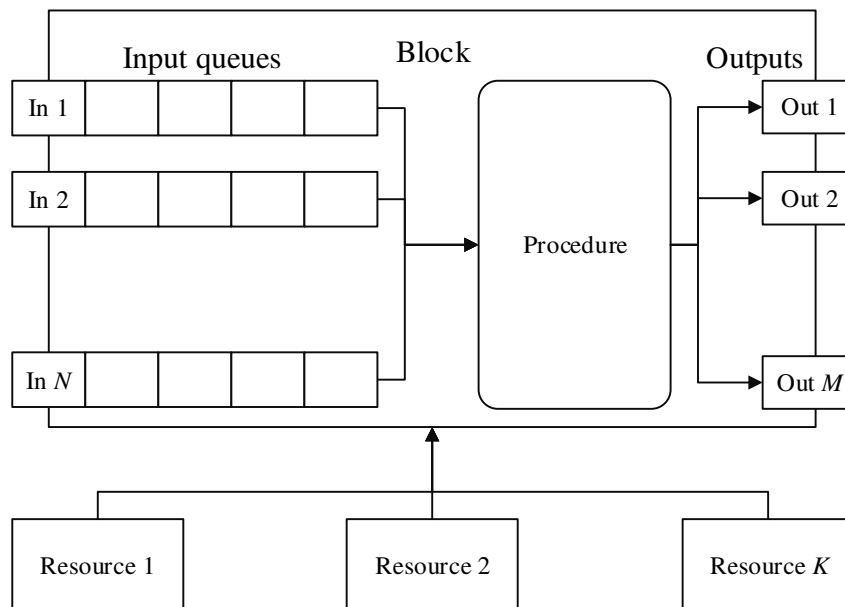


Рисунок 1 - Модель блока

Моделирование процесса с помощью сетей Петри происходит с помощью набора позиций и переходов сети. Последовательность перемещения маркеров из входных позиций перехода в выходные позиции образует моделируемый процесс. PRO-сети – это модификация сетей Петри, предназначенная для моделирования процессов. PRO -сети предоставляют возможности:

- детализации процесса;
- отображения параллельных процессов;
- расчёта временных задержек переходов;
- представления процесса на разных уровнях абстрагирования;
- отображения атрибутов, средствами которых можно учитывать качество.

PRO-сеть представляет пятерку вида:

$$\Phi = (T, L, R, V, M),$$

где T – конечное множество переходов (событий); L – множество позиций (состояний, условий), таких, что $L = \{ \dots, (l_i, \beta_i), \dots \}$ (l_i – позиция; β_i – максимальное число сообщений (меток, маркеров), которые допустимы для позиции L_i); R – отношение: $R = L \times T \cup T \times L$, т. е. позиции и переходы определяют множество дуг; V – конечное множество глобальных переменных v_i (переменных среды), доступное для всех процедур перехода π и процедур приоритета ρ . Если переменные среды $v_i \in V$ доступны процедурам π_i или ρ_i , то они не могут быть доступны процедурам π_j или ρ_j ($i \neq j$) до тех пор, пока существуют ограничения запуска перехода t_j . В качестве переменной среды могут выступать различные переменные, типы данных которых определяются тем языком, которым описаны процедуры перехода и приоритета; M – множество меток на элементах множества L ; $\mu_i \in M$ определяется как маркер на $l_i \in L$. Этот маркер указывает, что существует некоторое отмеченное условие. Любой маркер может иметь конечное число атрибутов: $\mu = \mu = \{ \alpha_1, \alpha_2, \dots, \alpha_n \}$ для $n \geq 0$, где атрибуты α_i имеют различные типы данных: целые, вещественные, логические (для языка процедур π_i и ρ_i).

Преимуществами PRO-сетей является возможность описания предусловий запуска переходов, рассмотрения временных особенностей моделирования процесса.

Переход $t \in T$ определяется как $(A, B, C, D, \rho, T, \Pi)$, где A и B – конечные множества входных дуг с нулем или более элементов; C и D – конечные множества выходных дуг с нулем или более элементов; ρ – процедура приоритета, которая

активируется при существовании входных условий для запуска перехода. С ее помощью можно:

- выбрать, но не модифицировать переменные среды и атрибуты меток ближайших входных позиций;
- выбрать входную позицию через дугу $b \in B$;
- выбрать процедуру $\pi \in \Pi$, связанную с переходом, и вызвать пробное выполнение;
- отсрочить запуск перехода, если необходимо, пока выходные условия (ограничения) не разрешат запуск перехода;
- T – множество задержек времени ($T \geq 0$) между инициализацией и завершением запуска перехода. Задержки времени $\tau \in T$ могут быть числом или выражением, которому может быть присвоено какое-либо значение.

T и Π связаны попарно (τ, π) . Процедура приоритета ρ может:

- выбирать любую переменную $v_i \in V$;
- читать атрибуты меток, расположенных в ближайших входных позициях;
- записывать значения в атрибуты меток, расположенных в ближайших выходных позициях;
- выбирать выходные позиции через дугу $d \in D$.

Выбор перехода представляет собой частный случай операции исключающего ИЛИ и определяется как

$$t_{bd} \in T : t_{bd} = (A, \{b\}, C, \{d\}, \rho, \{\tau\}, \{\pi\}),$$

где $b \in B$ – входная дуга, выбранная с помощью процедуры ρ ; $d \in D$ – выходная дуга, выбранная с помощью процедуры перехода π ; $\pi \in \Pi$ – процедура перехода, выбранная с помощью процедуры приоритета ρ ; $\tau \in T$ – задержка времени, связанная с выполнением процедуры π .

Графически переход изображается отрезком линии, если он является частью определенного далее элемента простой PRO-сети, и прямоугольником, если является частью определенного ниже элемента абстрактной PRO-сети (рис. 2). Позиция (условие) $l \in L$ может иметь максимальное число меток, отмеченное в β_i , где $0 \leq \beta_i \leq 1$ для выходных позиций в элементе PRO-сети; $0 \leq \beta_i \leq n$ для выходной позиции абстрактной PRO-сети (n – целое положительное число).

Позиция (условие) $l \in L$ может иметь максимальное число меток, отмеченное в β_i , где $0 \leq \beta_i \leq 1$ для выходных позиций в элементе PRO-сети; $0 \leq \beta_i \leq n$ для выходной позиции абстрактной PRO-сети (n – целое положительное число).

Для запуска двух видов переходов существуют следующие правила маркировки: l_a – для запуска перехода все позиции должны быть отмечены; l_b – хотя бы одна входная вершина должна быть отмечена; l_c – все выходные вершины должны быть помечены; l_d – хотя бы одна выходная вершина должна быть отмечена. PRO-сети обладают следующими свойствами: возможность детализации любого уровня вложенности; возможность организации обратных связей и параллельных процессов; расчет временных задержек переходов; учет качества проекта за счет использования атрибутов. Следовательно, PRO-сети адекватно отражают основные свойства процесса проектирования. С точки зрения системного подхода желателен простой и естественный переход от общесистемной модели (рис.3) к конструктивной. С этой целью воспользуемся формальным описанием стратегий и маршрутов проектирования [3].

Список литературы:

1. Шеннон Р. Имитационное моделирование систем - искусство и наука: Издательство: М.: Мир, 1978. 420 с.
2. Питерсон Дж. Теория сетей Петри и моделирование систем. М.: М.: Мир, 1984.
3. Goryachev A. V., Novakova N. E. System of basic operating units of the CAD process // 2017 XX IEEE International Conference on Soft Computing and Measurements (SCM). 2017. Vol. 1. P. 168 – 170
4. Горячев А. В., Новакова Н. Е. Модели и методы принятия проектных решений в распределённых САПР. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2019. 200 с.

УДК 004.94

НАДЕЖНОСТЬ КЛАСТЕРА ДУБЛИРОВАННЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ ПРИ ОБЕСПЕЧЕНИИ НЕПРЕРЫВНОСТИ ВЫЧИСЛИТЕЛЬНОГО ПРОЦЕССА

Богатырев Владимир Анатольевич

*д-р техн. наук, проф., профессор кафедры «Безопасность информационных систем»
ФГАОУ ВО «Санкт-Петербургский государственный университет аэрокосмического
приборостроения»*

Деркач Алексей Николаевич

аспирант

*ФГАОУ ВО «Санкт-Петербургский национальный исследовательский университет
информационных технологий, механики и оптики» (ИТМО)*

Аннотация. Предложена модель надежности кластерной системы, объединяющей дублированные вычислительные узлы. Модель учитывает ограничения на допустимое время прерывания вычислительного процесса при восстановлении дублированных узлов. Проанализированы возможности повышения надежности кластера реального времени при требовании сохранения непрерывности вычислительного процесса в результате миграции виртуальных машин.

Ключевые слова: дублирование, кластер, резервирование, восстановление, непрерывность вычислительного процесса.

Abstract. A model of the reliability of a cluster system that combines duplicated computing nodes is proposed. The model takes into account the restrictions on the allowed time of interrupting the computing process when restoring duplicate nodes. The possibilities of improving the reliability of a real-time cluster with the requirement of maintaining the continuity of the computing process as a result of the migration of virtual machines are analyzed.

Keywords: duplication, cluster, redundancy, recovery, continuity of the computing process.

Введение. Для киберфизических систем реального времени основополагающим является обеспечение их надежности и отказоустойчивости [1-4]. Надежность инфокоммуникационных средств достигается на основе использования дублированных компьютерных узлов с их объединением в отказоустойчивые кластеры. Отказоустойчивость кластерных систем, работающих в контуре управления физическими объектами, может обеспечиваться на основе технологий консолидации ресурсов их кластеризации и виртуализации, сопровождаемых репликацией и миграцией виртуальных машин между физическими серверами [5-9]. Миграция и репликация виртуальных машин ускоряет процесс реконфигурации после отказов физических ресурсов и способствует поддержке непрерывности вычислительного процесса,