

На правах рукописи

Петросов Давид Арегович

**ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ В ЗАДАЧАХ
КОНФИГУРИРОВАНИЯ ДИСКРЕТНЫХ ОБЪЕКТОВ С ЗАДАННЫМ
ПОВЕДЕНИЕМ**

Специальность 05.13.18 - Математическое моделирование,
численные методы и комплексы программ

АВТОРЕФЕРАТ
диссертации на соискание ученой степени
кандидата технических наук

Белгород – 2010

Работа выполнена в Белгородском государственном технологическом университете им. В.Г. Шухова

Научный руководитель: кандидат технических наук, доцент
Иванов Игорь Владимирович

Официальные оппоненты: доктор технических наук, профессор
Магергут Валерий Залманович

кандидат технических наук, доцент
Штифанов Андрей Иванович

Ведущая организация: Саратовский государственный технический университет

Защита диссертации состоится «30» сентября 2010 в 12:30 часов на заседании диссертационного совета Д212.014.06 при Белгородском государственном технологическом университете им. В.Г. Шухова, по адресу: 308012, г. Белгород, ул. Костюкова, 46.

С диссертацией можно ознакомиться в библиотеке Белгородского государственного технологического университета им. В.Г. Шухова.

Автореферат разослан «___» _____ 2010г.

Ученый секретарь
диссертационного совета



Ю.А. Бондаренко

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность работы. Дискретное модельное описание является характерным не только для технических, но и для технологических, организационных, экономических, биологических и других типов объектов, что делает целесообразным изучение общих подходов и методов разработки и исследования моделей для такого рода «дискретных» объектов (ДО). Свойством дискретности может обладать структура, поведение и время функционирования объектов.

Один из основных этапов создания ДО – этап конфигурирования, то есть определение состава элементов и межэлементных связей.

Критериями эффективности решения задачи конфигурирования являются качество полученного решения и затраты связанные с получением этого решения. Принятие решения при формировании конфигураций ДО осуществляется на основе генерации возможных структурных и функциональных моделей объекта, оценки этих моделей и выборе среди них лучшей альтернативы – такого варианта возможной модели ДО, которая функционирует должным образом, т.е. способна преобразовывать заданный входной сигнал в требуемый выходной сигнал.

В настоящее время имеются методы, позволяющие формализовать процедуру параметрического синтеза объекта, однако, отсутствуют эффективные методики, обеспечивающие структурное конфигурирование объекта с заданным поведением. Во многом это связано со сложностью моделей, учитывающих динамическое изменение связей в процессе функционирования объекта.

В основу работы положены три актуальных научных направления: имитационное моделирование, эволюционные методы и сети Петри.

Имитационное моделирование – один из мощных инструментов решения задач синтеза, который целесообразно использовать при конфигурировании объектов. Он дает возможность специалисту экспериментировать с проектируемыми и существующими объектами в случаях, когда делать это с реальным объектом невозможно или нецелесообразно. Разработке теоретических основ имитационного моделирования и применению этой методологии посвящены работы Н.Н. Моисеева, А.А. Самарского, В.П. Строгалева, Ю.И. Рыжикова, Р. Шеннона и др.

Эволюционные методы представляют собой современный математический аппарат нахождения эвристических решений сложных задач, основанный на понятии эволюции, что обуславливает их применение при конфигурировании ДО. Теоретические и прикладные вопросы применения эволюционных методов рассматривались в работах А.Г. Ивахненко, Н.И. Корсунова, В.М. Курейчика, Д. Э. Попова, Дж. Холанда, В. Ф. Хорошевского и др.

Сети Петри, как математический аппарат моделирования динамических дискретных систем, являются удобным формальным и графическим языком для моделирования систем с параллелизмом. Этот язык представляет собой обобщение теории автоматов, в котором может быть выражена концепция одновременно происходящих событий, что позволяет применять данный математический аппарат при создании сложных дискретных объектов. Описанию и применению данного математического аппарата посвящены работы В.Е. Котова, С.А. Юдицкого, В.З. Магергута, И.А. Ломазовой, Дж. Питерсона, и др.

Актуальность темы диссертационного исследования связана с необходимостью совершенствования методов разработки и исследования математических моделей сложных распределенных процессов функционирования дискретных технических и организационных систем.

Целью диссертационной работы является разработка и исследование имитационных моделей дискретных объектов с заданным поведением.

Поставленная цель определила следующие основные **задачи исследования:**

1. Выполнить анализ существующих математических моделей конфигурирования ДО;
2. Разработать имитационную модель формирования конфигурации ДО со статической структурой;
3. Разработать имитационную модель формирования конфигурации ДО с динамической структурой;
4. Выполнить программную реализацию конфигурирования ДО на основе разработанных математических моделей.

Методы исследований. Для решения поставленных задач использовались методы системного анализа, эволюционных вычислений, теории сетей Петри (СП) и дискретной математики.

Научная новизна диссертационной работы состоит в следующем:

1. Разработана имитационная модель конфигурирования дискретных объектов, учитывающая возможность динамической перестройки структуры, на основе математического аппарата вложенных сетей Петри и генетических алгоритмов;
2. Предложено представление моделей дискретных объектов в виде сетей Петри для кодирования генотипа в рамках генетических алгоритмов при решении задач конфигурирования;
3. Предложен подход к формированию критериев качества конфигурирования объектов, используемый в рамках генетического алгоритма для построения функции приспособленности.

Практическая значимость. На основе полученных моделей и алгоритмов разработана информационная система формирования конфигурации ДО с заданным поведением для решения задач обработки информации в произвольной предметной области. Практическая значимость подтверждается актами о внедрении отдельных результатов диссертационных исследований на ООО «Завод моющих средств» (г. Шебекино) и в Белгородском государственном технологическом университете им.В.Г.Шухова.

Положения, выносимые на защиту:

1. Способ построения конфигураций ДО с использованием генетических алгоритмов, описанных вложенными сетями Петри;
2. Метод поиска конфигурации дискретного объекта с фиксированными межкомпонентными связями;
3. Метод поиска конфигурации дискретного объекта с перестраиваемыми межкомпонентными связями;
4. Архитектура системы имитационного моделирования дискретных объектов на основе разработанных методов;
5. Комплекс проблемно-ориентированных программ имитационного моделирования дискретных объектов с заданным поведением.

Достоверность и обоснованность результатов исследований определяются использованием современных методов исследований, корректностью математических выводов и компьютерного моделирования, а так же результатами вычислительных экспериментов.

Личный вклад соискателя

Все разделы диссертационной работы выполнены лично автором. Все изложенные в диссертационной работе результаты исследований получены либо соискателем лично, либо при его непосредственном участии.

В работах, опубликованных в соавторстве, лично автором выполнена адаптация и формализация генетических алгоритмов на основе инструментария вложенных сетей Петри [4, 7, 8], разработана концепция построения структур ДО с фиксированными и перестраиваемыми межкомпонентными связями [5, 6, 9, 12], разработан вычислительный алгоритм и модуль поиска конфигурации [13, 14].

Апробация результатов диссертации. Основные научные и практические результаты докладывались и обсуждались на 7 Международном форуме «Радиоэлектроника и молодежь в XXI веке» (Харьков, 2003), 8 Международном форуме «Радиоэлектроника и молодежь в XXI веке» (Харьков, 2004), Международной научно-практической конференции «Информационные технологии в науку и образование» (Харьков 2005), Международной научно-технической конференции молодых учёных БГТУ им. В.Г. Шухова (г. Белгород, 2009), Международной научной конференции «Проблемы управления, передачи и обработки информации (АТМ-ТКИ-50)» (г. Саратов, 2009), XIV Международной научно-производственной конференции "Проблемы сельскохозяйственного производства на современном этапе и пути их решения" (г. Белгород, 2010), на ежегодных научно-технических семинарах кафедры «Информационные технологии» БГТУ им. В.Г.Шухова (2008-2010 г.г.).

Публикации. Основные положения работы изложены в 12 печатных работах, из которых 3 в изданиях, рекомендованных ВАК РФ по научной специальности диссертационной работы. Получено 2 свидетельства о государственной регистрации программ для ЭВМ.

Диссертационная работа состоит из введения, четырех глав, заключения и списка литературных источников из 139 наименований. Содержание работы изложено на 153 листах машинописного текста.

СОДЕРЖАНИЕ РАБОТЫ

Во **введении** обоснована актуальность темы диссертации, сформулирована цель и поставлены задачи, необходимые для ее достижения. Кратко изложены основные результаты, определяющие новизну и практическую ценность работы. Перечислены места реализации и апробации результатов работы.

Глава 1 «Обзор современного состояния проблемы» является вводно-постановочной, в ней поведен обзор современного состояния проблемной области, дано определение ДО, рассмотрены формальные и интеллектуальные методы конфигурирования ДО.

Рассмотрены объекты различной физической природы, как то: организационные, механические, электротехнические, технологические и объекты вычислительной техники. Показана общность этих объектов с точки зрения возможности их описания дискретными моделями с заданным поведением. Установлено существование класса задач конфигурирования дискретных объектов, часть из которых может быть решена путем использования имитационных моделей с фиксированными межэлементными связями, а часть потребует применения динамически перестраиваемых структур моделей.

Также проведен анализ используемых инструментариев и на основе этого анализа предложено использовать генетические алгоритмы как формальный метод поиска наилучшей конфигурации и сети Петри, как математический инструмент для построения имитационной модели компонентов ДО, а также как средство реализации самого генетического алгоритма. Проведен анализ существующих программных средств, способных решать поставленную задачу.

Глава 2 «Имитационная модель формирования конфигурации дискретного объекта со статической структурой» посвящена созданию имитационной математической модели процесса формирования конфигурации ДО с заданным поведением, имеющего фиксированный набор связей.

Для этого описывается объект O

$$O = \langle S, C \rangle, \quad (1)$$

где O – объект, который необходимо сконфигурировать, S – фиксированная структура объекта, C – состав объекта.

$$C = (C_1, \dots, C_i, \dots, C_R), \quad (2)$$

где C_i – i -й компонент объекта, R – количество компонентов объекта.

$$C_i = \left\{ C_{ij} \right\}_{j=1}^{M_i}, \quad (3)$$

где C_{ij} – j -й экземпляр i -го компонента, M_i – количество экземпляров i -го компонента.

$$P = \left\{ P_k \right\}_{k=1}^L, \quad (4)$$

где P – множество свойств, которыми может обладать конфигурируемый объект, P_k – k -е свойство множества P , L – количество свойств множества P .

Требуется для заданного свойства P_{k_0} , которым должен обладать объект O , подобрать по одному экземпляру $C_{ij(k_0)}$ каждого компонента C_i так, чтобы объект

$$O = \left\langle S, \left(C_{1j(k_0)}, \dots, C_{Rj(k_0)} \right) \right\rangle, \quad (5)$$

обладал свойством P_{k_0} .

Для описания объекта O (1) в виде сети Петри (СП), необходимо определять его структуру S и состав C также на языке СП.

Согласно (2) и (3) состав C полностью определяется компонентами, поэтому каждый компонент C_i должен быть описан в виде СП, а это в свою очередь означает, что каждый экземпляр компонента C_{ij} необходимо определять на языке СП.

Каждый C_{ij} экземпляр обозначим через СП, моделирующую его, – PN_{ij} (Petri Nets) $_{ij}$. Выделяем у каждого экземпляра входы и выходы, которые моделируются позициями сети PN_{ij} . Множество входных и выходных позиций обозначаются через IN_{ij} и OUT_{ij} .

Между элементами множеств IN_{ij} и IN_i , а также OUT_{ij} и OUT_i должны существовать взаимно однозначные соответствия $F_{IN}^{ij}: IN_{ij} \rightarrow IN_i$ и $F_{OUT}^{ij}: OUT_{ij} \rightarrow OUT_i$. На рис. 1 (а) показано соответствие между компонентами и экземплярами.

СП, моделирующая компонент C_i , обозначается через PN_i (Petri Nets) $_i$. Если компонент C_i , представлен экземпляром C_{ij_0} , то $PN_i = PN_{ij_0}$.

Структура S объекта O полностью определяется взаимосвязями между компонентами, которые можно моделировать переходами СП. Обозначим множество таких переходов через T .

Каждый переход $t_q \in T$ соединяет выходные позиции некоторых компонентов $C_{i_1}, C_{i_2}, \dots, C_{i_A}$ с входными позициями некоторых других компонентов $C_{i_{A+1}}, C_{i_{A+2}}, \dots, C_{i_{A+B}}$, что наглядно иллюстрируется рис.1 (б).

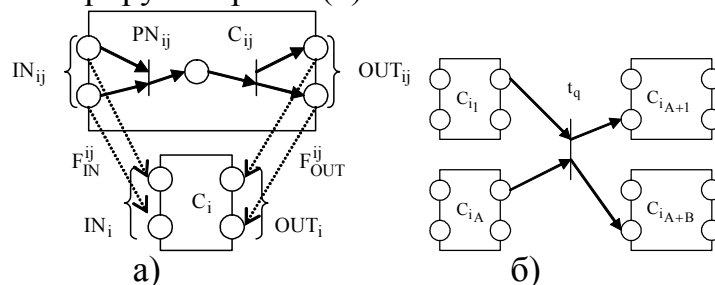


Рисунок 1. Соответствие (а) и связи (б) между компонентом и экземпляром

Соответствие $F: T \rightarrow \bigcup_{i=1}^R (IN_i \cup OUT_i)$ полностью определяет структуру S объекта O .

СП, моделирующая объект O , описывается в виде кодовой строки

$$PN = \langle PN_1, \dots, PN_i, \dots, PN_R, T, F \rangle, \quad (6)$$

где PN_i – модель компонента C_i , а множество переходов T и соответствие F определяют структуру S объекта O .

Для выбора необходимой модели объекта O определим поведение объекта как способность преобразовывать заданный входной сигнал в необходимый выходной. Поведение объекта описывается парой неотрицательных целочисленных векторов

$$Z_{IN} = (z_1^{IN}, \dots, z_{V_0}^{IN}) \quad \text{и} \quad Z_{OUT} = (z_1^{OUT}, \dots, z_{W_0}^{OUT})$$

где z_v^{IN} – число меток, поступивших в v -ю входную позицию перед запуском сети PN , z_w^{OUT} – число меток, появившихся в w -й выходной позиции после остановки сети PN , V_0 и W_0 – число элементов множеств IN и OUT соответственно.

Таким образом, поставленная задача сводится к следующей. Среди всех гипотетически возможных моделей PN объекта O найти такую, которая обладает свойством Z_k .

Для того чтобы проверить, обладает ли модель PN свойством Z_k , необходимо сформировать эту модель, на ее вход IN подать вектор Z_{IN}^k , запустить сеть PN и после ее остановки сравнить метки на выходе OUT с вектором Z_{OUT}^k .

Меру близости будем определять, привлекая понятие метрического пространства и рассматривая полученный вектор Z_{OUT} и эталонный вектор Z_{OUT}^k как элементы евклидова пространства \mathfrak{R}^{W_0} – множества упорядоченных наборов из W_0 действительных чисел $x = (x_1, \dots, x_{W_0})$ с расстоянием

$$\rho_1(x, y) = \sum_{w=1}^{W_0} |x_w - y_w| \quad (7)$$

где $y = (y_1, \dots, y_{W_0})$.

Чем меньше $\rho_1(Z_{OUT}, Z_{OUT}^k)$, тем ближе модель PN к свойству Z_k . Если $\rho_1(Z_{OUT}, Z_{OUT}^k) = 0$, то модель PN обладает свойством Z_k . Естественно рассматривать расстояние ρ_1 как целевую функцию.

Для поиска конфигурации используется генетический алгоритм (ГА). Операторы ГА будут моделироваться переходами вложенных сетей Петри (ВСП), а генотипы – макромметками ВСП.

Множество генотипов, непосредственно обрабатываемое ГА, называется популяцией. Начальная популяция может формироваться либо автоматически (например, случайным выбором экземпляров компонентов), либо задаваться непосредственно разработчиками ДО. Обозначим начальную популяцию через

$$G^0 = (PN^1, \dots, PN^{2n}),$$

где PN^i – i -ая модель объекта проектирования в виде СП, $2n$ – фиксированный размер популяции. Здесь для простоты дальнейшего описания предполагается, что размер популяции является четным числом.

Каждую модель PN^i разместим в соответствующей позиции A_i в качестве макромметки ВСП.

Оператор отбора моделируется переходом SEL , который будет копировать PN^i из позиции A_i и размещать эти копии в позициях $B_1, B_2, \dots, B_{2i-1}, B_{2i}, \dots, B_{2n-1}, B_{2n}$ в качестве макромметок ВСП. Обозначим макромметки, размещенные в позициях B_{2n-1} и B_{2i} , через $PN^{j_{2i-1}}$ и $PN^{j_{2i}}$. Макромметки размещаются в позициях B_1, \dots, B_{2n} в соответствии со значением целевой функции (7) по правилу: чем меньше значение функции (7), тем меньше номер позиции, в которой размещается макромметка. Таким образом, в позиции B_1 размещается самая лучшая макромметка, в позиции B_2 – чуть худшая макромметка и т.д. до позиции B_{2n} , в которой будет находиться самая плохая макромметка.

Оператор скрещивания моделируется переходом $CROSS_i$, который будет забирать макромметки $PN^{j_{2i-1}}$ и $PN^{j_{2i}}$ из позиций B_{2i-1} и B_{2i} , скрещивать их и размещать новые

макрометки $PN_*^{j_{2i-1}}$ и $PN_*^{j_{2i}}$ в позициях D_{2i-1} и D_{2i} соответственно. Макрометки скрещиваются по следующему правилу:

1. Случайным образом из множества $\{1,2,\dots,R\}$ выбирается некоторое число, которое мы обозначим через r .

2. В генотипах $PN_*^{j_{2i-1}}$ и $PN_*^{j_{2i}}$ выделяем модель r -го компонента

$$PN_*^{j_{2i-1}} = \langle PN_1^{j_{2i-1}}, \dots, PN_r^{j_{2i-1}}, \dots, PN_R^{j_{2i-1}}, T, F \rangle \quad \text{и} \quad PN_*^{j_{2i}} = \langle PN_1^{j_{2i}}, \dots, PN_r^{j_{2i}}, \dots, PN_R^{j_{2i}}, T, F \rangle$$

3. Меняем местами выделенные модели и получаем новые макрометки

$$PN_*^{j_{2i-1}} = \langle PN_1^{j_{2i-1}}, \dots, PN_r^{j_{2i}}, \dots, PN_R^{j_{2i-1}}, T, F \rangle \quad \text{и} \quad PN_*^{j_{2i}} = \langle PN_1^{j_{2i}}, \dots, PN_r^{j_{2i-1}}, \dots, PN_R^{j_{2i}}, T, F \rangle$$

Оператор мутации моделируется переходом MUT_{2i} , который будет забирать макрометку $PN_*^{j_{2i}}$ из позиции D_{2i} , осуществлять ее мутацию и размещать новую макрометку $PN_{**}^{j_{2i}}$ в позиции E_{2i} . Макрометка подвергается мутации по следующему правилу:

1. Случайным образом из множества $\{1,2,\dots,R\}$ выбирается некоторое число, которое мы обозначим через r .

2. В генотипе $PN_*^{j_{2i}}$ выделяем модель r -го компонента $PN_*^{j_{2i}} = \langle PN_1^{j_{2i}}, \dots, PN_r^{j_{2i}}, \dots, PN_R^{j_{2i}}, T, F \rangle$

3. Из множества $\{PN_{rj}\}_{j=1}^M$ моделей экземпляров r -го компонента случайным образом выбираем новую модель, которую мы обозначим через $PN_{r_{new}}^{j_{2i}}$, заменяем ею

модель $PN_r^{j_{2i}}$ и получаем новую макрометку $PN_{**}^{j_{2i}} = \langle PN_1^{j_{2i}}, \dots, PN_{r_{new}}^{j_{2i}}, \dots, PN_R^{j_{2i}}, T, F \rangle$

Оператор мутации макрометки в позиции D_{2i-1} моделируется аналогично.

Оператор редукции моделируется переходом RED, который будет забирать макрометки из позиций E_1, \dots, E_{2n} и из позиций A_1, \dots, A_{2n} , удалять худшие макрометки, а оставшиеся размещать в позициях A_1, \dots, A_{2n} . Макрометки удаляются по следующему правилу:

1. Для каждой макрометки вычисляется целевая функция (7).

2. В позиции A_1 размещается самая лучшая макрометка, в позиции A_2 – чуть худшая макрометка и т.д. до позиции A_{2n} , в которой будет находиться самая плохая макрометка. Обозначим макрометку, размещенную в позиции A_i , через PN_*^i .

3. Все остальные макрометки удаляются. Заметим, что будет удалено ровно $2n$ макрометок.

Структура вложенной СП с обозначением всех макрометок показана на рис. 2.

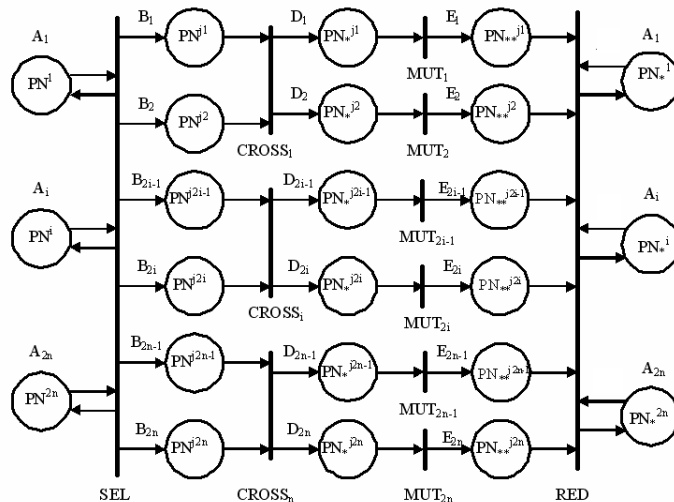


Рисунок 2. Формализация генетического алгоритма с помощью ВСП

Глава 3 «Имитационная модель формирования конфигурации дискретного объекта с динамической структурой» посвящена созданию имитационной математической модели конфигурирования ДО с переменными межкомпонентными связями.

Рассматривается класс задач формирования конфигурации ДО, который можно представить короткежем:

$$O = \langle In, Out, \{S_k\}_{k=1}^K, \{f_a\}_{a=1}^A, \{F_b\}_{b=1}^B \rangle, \quad (8)$$

где O – объект, который необходимо сконфигурировать;

In – множество входных данных объекта O ;

Out – множество выходных данных объекта O ;

S_k – k -я подсистема объекта O ;

f_a – функция, определяющая, какие выходные данные соответствуют входным данным: $f_a : In \rightarrow Out$;

F_b – бинарное отношение на множестве $\{S_k\}_{k=1}^K : F_b \subset \{S_k\}_{k=1}^K \times \{S_k\}_{k=1}^K$.

Требуется для заданной функции f_{a_0} подобрать бинарное отношение F_{b_0} такое, чтобы множество подсистем $\{S_k\}_{k=1}^K$ обеспечивало обработку объектом O входных данных в выходные данные в соответствии с функцией f_{a_0} .

Входы объекта O будут моделироваться множеством позиций $P_{in} = \{p_m^{in}\}_{m=1}^M$, где M – количество входов объекта, а ее выходы – множеством позиций $P_{out} = \{p_n^{out}\}_{n=1}^N$, где N – количество выходов объекта (рис. 3).

Входы подсистемы S_k – это множество $P_k^{in} = \{p_{k,m}^{in}\}_{m=1}^{M(k)}$, где $M(k)$ – количество входов подсистемы S_k , а ее выходы – множество $P_k^{out} = \{p_{k,n}^{out}\}_{n=1}^{N(k)}$, где $N(k)$ – количество выходов подсистемы S_k .

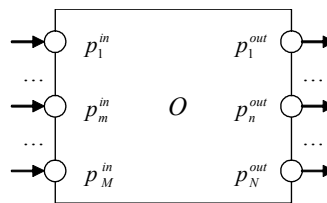


Рисунок 3. Контекстная модель ДО в задаче конфигурирования

Входы и выходы подсистем объекта O моделируются множеством позиций. Подсистемы $\{S_k\}_{k=1}^K$ могут быть связаны как между собой, так и с входами и выходами объекта O . Эти связи будут моделироваться множеством переходов $T = \{t_q\}_{q=1}^Q$, где Q – количество переходов. Каждый переход t_q характеризуется своими входными и выходными позициями. Входами перехода t_q могут быть как любые входы объекта O , так и любые выходы подсистем $\{S_k\}_{k=1}^K$. Выходами перехода t_q могут быть как любые выходы объекта O , так и любые входы подсистем $\{S_k\}_{k=1}^K$. Обозначим входы перехода t_q через In_q , а выходы – через Out_q . Тогда, исходя из вышесказанного, $In_q \subset P_{in} \cup \left(\bigcup_{k=1}^K P_k^{out} \right)$ и $Out_q \subset P_{out} \cup \left(\bigcup_{k=1}^K P_k^{in} \right)$.

Для каждого входа объекта O должен существовать переход t_q , соединенный с этим входом, а для каждого выхода каждой подсистемы S_k должен существовать переход t_q , соединенный с этим выходом. Формально эти требования можно записать в виде равенства $\bigcup_{q=1}^Q In_q = P_{in} \cup \left(\bigcup_{k=1}^K P_k^{out} \right)$. Соответственно, для каждого выхода объекта O должен существовать переход t_q , соединенный с этим выходом, и для каждого входа каждой подсистемы S_k должен существовать переход t_q , соединенный с этим входом. Формально эти требования запишем в виде равенства $\bigcup_{q=1}^Q Out_q = P_{out} \cup \left(\bigcup_{k=1}^K P_k^{in} \right)$.

Разные переходы могут иметь некоторые общие входные и выходные позиции, но полностью совпадающих переходов быть не должно. Это требование формализуется следующим образом: $In_{q1} = In_{q2} \Rightarrow Out_{q1} \neq Out_{q2}$ (при $q1 \neq q2$) и $Out_{q1} = Out_{q2} \Rightarrow In_{q1} \neq In_{q2}$ (при $q1 \neq q2$).

Множество подсистем $\{S_k\}_{k=1}^K$ вместе с подмножеством множества переходов $T = \{t_q\}_{q=1}^Q$ полностью определяют текущую структуру объекта O . Не обязательно все подсистемы $\{S_k\}_{k=1}^K$ должны быть задействованы в текущей структуре. При этом перестройка структуры объекта O определяется изменением подмножества задействованных в данный момент переходов.

Перестройка функций происходит в подсистемах $\{S_k\}_{k=1}^K$. В общем случае каждой подсистеме S_k можно сопоставить множество сетей Петри, которые будут моделями программ обработки данных. Это множество можно представить как PN_k и определить его следующим образом: $PN_k = \{PN_{k,r}\}_{r=1}^{R(k)}$, где $PN_{k,r}$ – r -й алгоритм обработки данных подсистемой S_k , представленный в виде СП.

Формально необходимо для каждой СП $PN_{k,r}$ описать ее позиции, переходы и дуги. В общем случае сеть $PN_{k,r}$ моделирует некоторое действие по преобразованию входных данных в выходные данные. Поэтому в качестве моделей алгоритмов будут рассматриваться не СП $PN_{k,r}$, а переходы («вырожденные» сети Петри). Таким образом, множество PN_k будет выглядеть так $PN_k = \{t_{k,r}\}_{r=1}^{R(k)}$. Входами $In_{k,r}$ каждого перехода $t_{k,r}$ могут любые входы подсистемы S_k : $In_{k,r} \subset P_k^{in}$, а выходами $Out_{k,r}$ каждого перехода $t_{k,r}$ могут любые выходы подсистемы S_k : $Out_{k,r} \subset P_k^{out}$.

Модель объекта O в виде СП может выглядеть так, как показано на рис. 4. В этой модели выделен путь, по которому входные данные, поступив на верхние два входа объекта O (на рисунке в виде двух меток), проходят через ее подсистемы и появляются на нижнем выходе объекта O . Пунктиром выделены недействующие переходы и дуги.

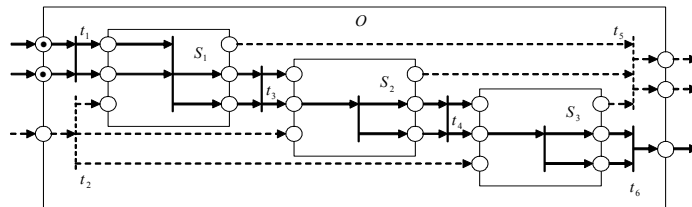


Рисунок 4. Пример имитационной модели ДО на основе сети Петри

Формализуем генотип G , он должен определять:

– какие переходы из множества $T = \{t_q\}_{q=1}^Q$ будут представлены в модели объекта O ;

– какие переходы из множества $PN_k = \{t_{k,r}\}_{r=1}^{R(k)}$ будут представлены в модели каждой подсистемы S_k .

Таким образом, генотип G будет иметь следующий вид: $G = (g_1, \dots, g_q, \dots, g_Q, h_1, \dots, h_k, \dots, h_K)$, где $g_q \in \{0, 1\}$ и $h_k \in \{0, 1, \dots, r, \dots, R(k)\}$.

Если $g_q = 0$, то переход t_q отсутствует в модели объекта O , иначе ($g_q = 1$) переход t_q присутствует в модели объекта O . Если $h_k = 0$, то в модели подсистемы S_k отсутствует алгоритм обработки данных. Если $h_k = r$ ($r \in \{1, \dots, R(k)\}$), то в модели подсистемы S_k представлен переход $t_{k,r}$.

Среди всех гипотетически возможных моделей объекта O необходимо найти такую, которая решает поставленную задачу: из имеющегося множества подсистем $\{S_k\}_{k=1}^K$ и системы переходов $T = \{t_q\}_{q=1}^Q$ построить такой объект O , который на входной вектор реагировал бы требуемым выходным вектором.

Для оценки этой близости вводим обозначение для требуемого вектора: $V = (v_1, \dots, v_n, \dots, v_N)$, где v_n – количество меток, которые должны находиться в позиции p_n^{out} объекта O . Вектор, который получился в результате работы объекта O , обозначим

через $W=(w_1, \dots, w_n, \dots, w_N)$, где w_n – количество меток, которые реально находятся в позиции p_n^{out} объекта O . Разницу (расстояние) между требуемым и реальным вектором можно оценивать по формуле $\rho(V, W) = \sum_{n=1}^N |v_n - w_n|$.

Выбор целевой функции влияет на эффективность работы генетического алгоритма. Поэтому на практике можно использовать и формулу, задающую классическое евклидово расстояние $\rho(V, W) = \sqrt{\sum_{n=1}^N (v_n - w_n)^2}$, и редко используемую формулу $\rho(V, W) = \max_{1 \leq n \leq N} |v_n - w_n|$. При программной реализации генетического алгоритма необходимо заложить в систему возможность работы с формулой $\rho(V, W) = \left(\sum_{n=1}^N |v_n - w_n|^p \right)^{1/p}$, которая при $p=1$, $p=2$ и $p \rightarrow \infty$ представляет собой рассмотренные выше формулы. Чем меньше ρ , тем ближе модель объекта O к требуемой конфигурации. При $\rho=0$ модель полностью соответствует заданному требованию. После выбора целевой функции можно описать операторы генетического алгоритма.

Оператор скрещивания можно описать следующим образом. Рассмотрим два генотипа: $G_1 = (g_1^1, \dots, g_q^1, g_{q+1}^1, \dots, g_Q^1, h_1^1, \dots, h_k^1, h_{k+1}^1, \dots, h_K^1)$, $G_2 = (g_1^2, \dots, g_q^2, g_{q+1}^2, \dots, g_Q^2, h_1^2, \dots, h_k^2, h_{k+1}^2, \dots, h_K^2)$.

Случайным образом выбираем два числа: q из диапазона (множества) $\{1, 2, \dots, Q\}$ и k из диапазона $\{1, 2, \dots, K\}$, а затем меняем соответствующие участки генотипов. Таким образом, из родителей G_1 и G_2 получаются потомки G_3 и G_4 , наследующие свойства родителей: $G_3 = (g_1^1, \dots, g_q^1, g_{q+1}^2, \dots, g_Q^2, h_1^1, \dots, h_k^1, h_{k+1}^2, \dots, h_K^2)$, $G_4 = (g_1^2, \dots, g_q^2, g_{q+1}^1, \dots, g_Q^1, h_1^2, \dots, h_k^2, h_{k+1}^1, \dots, h_K^1)$.

Можно ограничиться и выбором какого-то одного числа (q или k). Сколько чисел будет выбрано (одно или два) и, если одно, то какое именно, можно также определять случайным образом. Можно также выбрать для каждого диапазона ($\{1, 2, \dots, Q\}$ и $\{1, 2, \dots, K\}$) по две точки и меняться «серединками».

Оператор мутации описывается следующим образом. Рассмотрим один генотип $G=(g_1, \dots, g_q, \dots, g_Q, h_1, \dots, h_k, \dots, h_K)$. Случайно выбираем два числа: q из диапазона $\{1, 2, \dots, Q\}$ и k из диапазона $\{1, 2, \dots, K\}$. А затем меняем значения g_q и h_k следующим образом. Если было $g_q=0$, меняем его на единицу: $g_q=1$. И наоборот: если было $g_q=1$, то станет $g_q=0$. Если было $h_k=r$, то меняем его на любое другое из диапазона $\{0, 1, \dots, r-1, r+1, \dots, R(k)\}$. Естественно, при $r=0$ это будет диапазон $\{1, \dots, R(k)\}$, а при $r=R(k)$ это будет диапазон $\{0, 1, \dots, R(k)-1\}$.

Оператор редукции удаляет слабые генотипы. Для этого после скрещивания определяются качества всех потомков. После чего потомки и родители объединяются в одно множество генотипов, что позволяет упорядочить их по качеству. В соответствии с настройками ГА происходит отсеивание конфигураций.

В частном случае перестройка структуры объекта O может происходить с использованием межкомпонентной шины.

Вся система элементов ДО и шин обрабатывается последовательно от входа системы к ее выходу. Сначала первый слой элементов получает и обрабатывает входные сигналы. Затем по первой шине сигналы передаются на второй слой. После обработки сигналы по второй шине поступают на третий слой и т.д. Процесс заканчивается после обработки сигналов последним слоем элементов, и выходной вектор сравнивается с искомым.

Формально, для каждой шины задано множество входных позиций $In = \{In_m\}$ и множество выходных позиций $Out = \{Out_m\}$ ($m = 1, 2, \dots, M$). Входные позиции – это выходы предыдущего блока элементов ДО, а выходные позиции – входы последующего блока элементов (рис. 5).

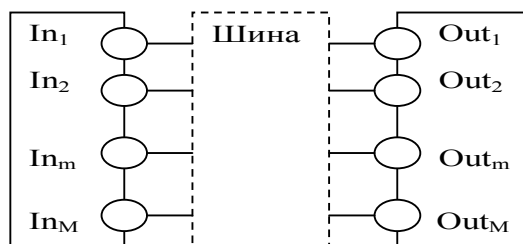


Рисунок 5. Обобщение межкомпонентных связей в слой «Шина»

Для моделирования межкомпонентной шины целесообразно использовать: связи («вырожденные сети Петри»), элементы ДО и логические элементы. Каждой выходной позиции Out_m сопоставляются (генерируются случайным образом):

- блок ИЛИ с указанием множества входных позиций;
- блок ИЛИ-НЕ с указанием множества входных позиций;
- количество блоков И;
- количество блоков И-НЕ;
- множество входных позиций для каждого блока И (И-НЕ).

Выходной позиции могут быть сопоставлены один блок ИЛИ, один блок ИЛИ-НЕ, несколько блоков И, несколько блоков И-НЕ.

Шина работает следующим образом. Выходные позиции обрабатываются по порядку от Out_1 до Out_m согласно алгоритму. Первым проверяется блок ИЛИ, затем ИЛИ-НЕ. Потом проверяются блоки И, а в самом конце – И-НЕ. Для ускорения процесса сначала проверяются блоки И с меньшим количеством входных позиций. Проверка останавливается, когда найден блок, который можно будет запустить.

Таким образом, шина полностью характеризуется набором блоков, сопоставленных каждой выходной позиции. Поэтому скрещивание шин – это обмен выходными позициями.

Родители: X-шина: $xOut_1, \dots, xOut_m, xOut_{m+1}, \dots, xOut_M$
 Y-шина: $yOut_1, \dots, yOut_m, yOut_{m+1}, \dots, yOut_M$
 Потомки: X-шина: $xOut_1, \dots, xOut_m, yOut_{m+1}, \dots, yOut_M$
 Y-шина: $yOut_1, \dots, yOut_m, xOut_{m+1}, \dots, xOut_M$

Мутация шины – изменение для какой-либо выходной позиции:

- множества входных позиций блока ИЛИ;
- количества блоков И;
- множества входных позиций для каждого блока И.

В главе 4 «Разработка системы имитационного моделирования конфигурирования дискретного объекта» для анализа эффективности и проверки адекватности полученных математических моделей процесса формирования конфигураций ДО создана информационная система (ИС), архитектура которой представлена на рис. 6.

Архитектура включает 5 основных элементов: элементная база, база знаний, генетический алгоритм на основе вложенных сетей Петри, модуль поиска и модуль анализа моделей. ИС выполняет функцию поиска конфигурации ДО, на основе моделей ДО хранящихся в элементной базе формализованных сетями Петри. Так же ИС помогает анализировать полученные конфигурации ДО, соответствующих критериям поиска, используя базу знаний (БЗ). Модуль эмуляции и редактирования, входящий в состав блока анализа, позволяет оценить и отредактировать модели приближенные к критериям поиска, и принять решение о вводе новых элементов ДО в элементную базу.

Функции модельера разделены между двумя интерфейсными модулями интерфейсом «Пользователя» и интерфейсом «Эксперта». Интерфейс «Пользователя» предназначен для специалиста в предметной области, в которой происходит моделирование. Интерфейс позволяет управлять системой с помощью параметров поиска, элементной базы и работой с модулем анализа. Интерфейс «Эксперта»

предназначен для работы с настройками генетического алгоритма, базы знаний и элементной базы. Тем самым пользователь ИС должен быть специалистом лишь в своей предметной области, а настройки конфигурирования ДО проводятся экспертом.

На основе функциональной модели ИС была осуществлена ее программная реализация. Комплекс проблемно-ориентированных программ разработан на языке программирования Object Pascal (в среде Delphi) и зарегистрирован в ФГУ ФИПС. Из списка функциональных задач были решены следующие:

- Генерация генотипов по заданным настройкам;
- Эмуляция генотипов;
- Подбор структуры сети по заданному входу и выходу;
- Модуль для тестирования для проведения экспериментов;
- Менеджер генотипов.

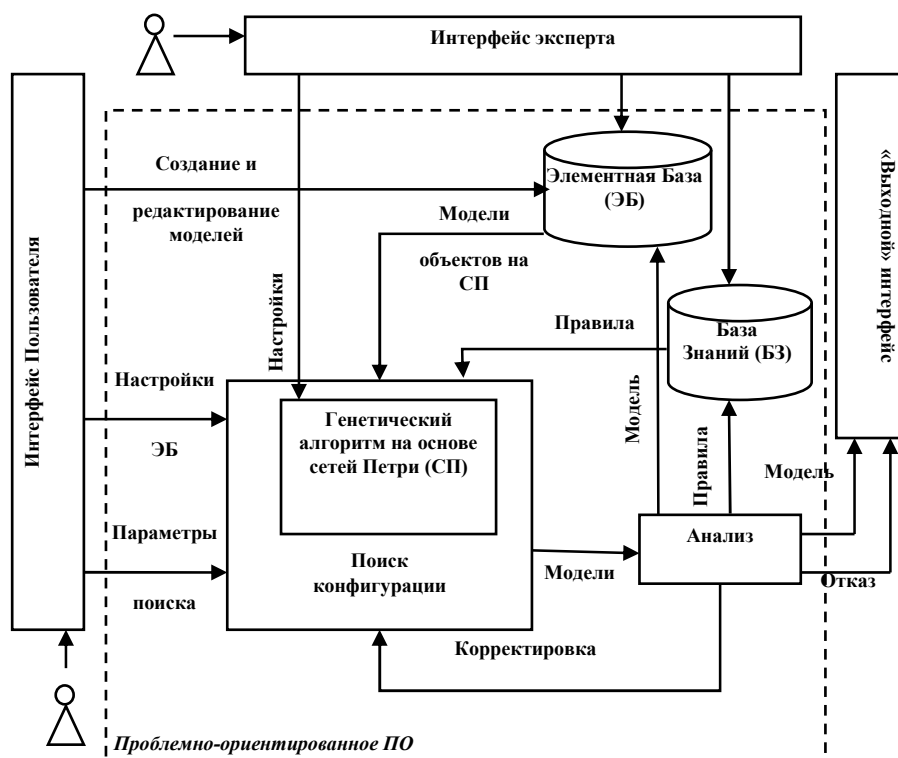


Рисунок 6. Архитектура информационной системы моделирования ДО

Входными документами в системах являются файлы типовых устройств описанных в виде сетей Петри. Для стандартизации предусмотрено описание сетей Петри при помощи языка PNML (Petri Net Markup Language).

На основе предложенного метода имитационного моделирования ДО и разработанной архитектуры ИС реализован и зарегистрирован комплекс программ. Показана его работа в процессе решения задачи формирования конфигурации вычислительных систем с заданным поведением, задачи распределения ресурсов на примере формирования расписания занятий в вузе и построена система массового обслуживания операционного зала банка. Среди иных задач использования предложенного метода можно назвать, например, задачу нахождения ветви, характеризующую правильную СП, на дереве достижимых маркировок, нахождение нужных конфигураций при создании технического комплекса АСУ ТП и ряд других.

Для решения задачи поиска конфигурации триггерного устройства построены математические модели триггеров и их конфигурации на основе сетей Петри; формирования конфигурации проводится на основе адаптированного генетических алгоритмов.

Модель конфигурации триггерного устройства с «квадратной» структурой 2×2 показана на рис. 7.

Если подсистема $S_{1,1}$ моделируется RS-триггером, подсистема $S_{1,2}$ – T-триггером, $S_{2,1}$ – D-триггером, а подсистема $S_{2,2}$ – T-триггером, и все триггеры находятся в единичном состоянии, то конфигурация $O=(RS, T, D, T)$ будет иметь такой вид, как показано на рисунке 8 в соответствии с (6).

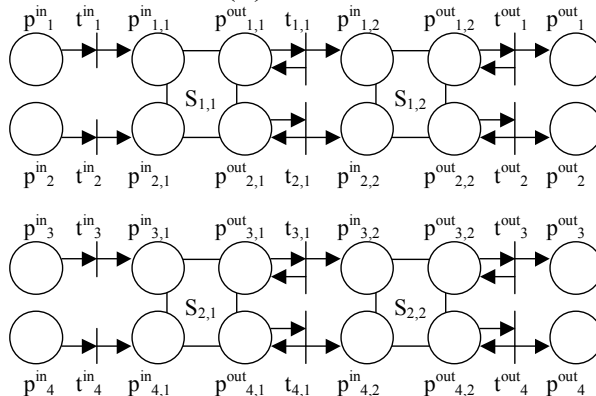


Рисунок 7. Модель конфигурации ДО размерностью 2×2

Конфигурация $O=(RS, T, D, T)$ работает по следующему алгоритму:

- на вход подаются данные, моделируемые метками;
- запускаются «входные» переходы $t^{in}_1 - t^{in}_4$;
- запускаются триггеры «первой очереди» $RS_{1,1}$ и $D_{2,1}$;
- запускаются переходы «первой очереди» $t_{1,1} - t_{4,1}$;
- запускаются триггеры «второй очереди» $T_{1,2}$ и $T_{2,2}$;
- запускаются «выходные» переходы $t^{out}_1 - t^{out}_4$;
- в результате получается выходной вектор.

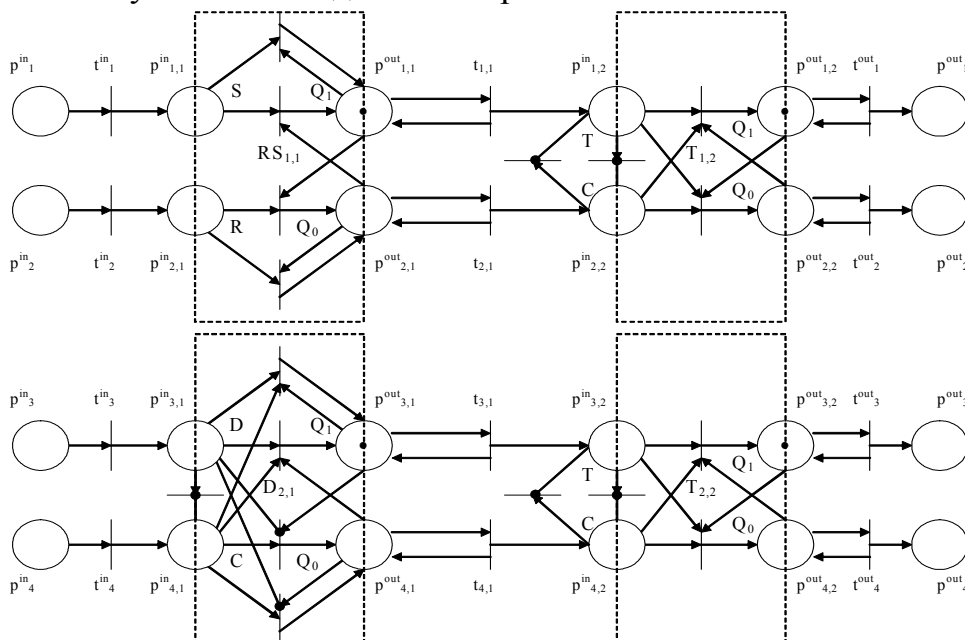


Рисунок 8. Пример конфигурация триггерного устройства с начальным состоянием (1, 1, 1, 1)

Например, если на вход конфигурации подать вектор (0, 1, 0, 1), то на выходе получится вектор (1, 0, 1, 0).

Задача составления расписания занятий представлена в виде ситуации распределения ресурсов, где ресурсами являются студенческие группы, а средствами обработки аудиторные помещения.

Для решения задачи распределения ресурсов, поступающих во входные позиции IN (8) дискретного объекта структура генотипа представлена с использованием топологии «звезда» в следующем виде (рис. 9).

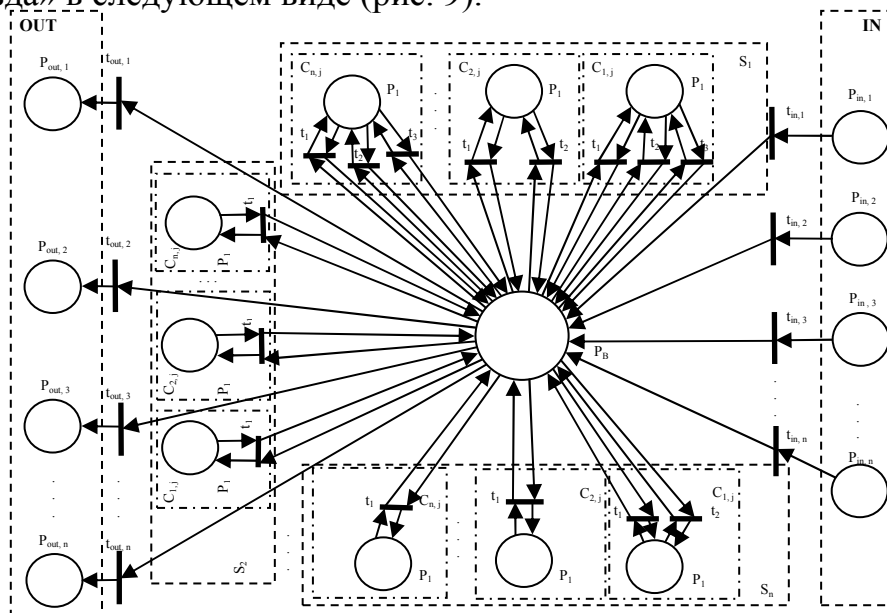


Рисунок 9. Структура генотипа в задаче распределения ресурсов

Объект O состоит из множества подсистем $O=(S_1, \dots, S_n)$, которые в свою очередь состоят из компонентов $C=(C_1, \dots, C_i)$ и экземпляров $C_i=\{C_{ij}\}_{j=1}^m$.

Позиция P_B – является источником, в который поступают ресурсы для обработки. Количество переходов в модели C_{ij} должно соответствовать количеству ресурсов, которое экземпляр обрабатывает за одну итерацию.

Метка во втором уровне вложенной сети Петри является ресурсом и представлена следующим образом $K=\langle G, D, M \rangle$,

где $M=(M_1, \dots, M_n)$ – маркер обработки ресурса;

$G=(G_1, \dots, G_n)$ – маркер принадлежности к типу ресурсов;

$D=(D_1, \dots, D_n)$ – маркер принадлежности к совместному использованию.

В задаче составления расписания маркер принадлежности к совместному использованию D – индикатор потока для проведения лекционных занятий (лекционное занятие может проводиться более чем с одной группой).

Маркер принадлежности к типу ресурсов G – индикатор специальности (ввиду того, что предметы на разных специальностях не всегда совпадают по своему наполнению материалом).

Маркер обработки ресурса M – состоит из количества и типа аудиторных занятий на учебный день.

Количество циклов обработки ресурсов в предложенной модели оценивается:

$$\max_{r=1, R} \left(\max_{i=1, N(r)} |M_{r,i}| + r \right), \text{ где } |M_{r,i}| \text{ – мощность } i\text{-го ресурса в } IN_r.$$

Модель системы массового обслуживания операционного зала банка описана в диссертационной работе.

Оценка эффективности работы ГА на ВСП осуществлялась экспериментально, на примере конфигурирования ДО различной размерности со статической и динамической структурой, с помощью разработанного и зарегистрированного комплекса проблемно-ориентированных программ в сравнении с методом прямого перебора и методами направленного перебора (ветвей и границ, золотое сечение). Полученные результаты приведены в диссертационной работе.

В таблице 1 представлено среднее время работы адаптированного генетического алгоритма при нахождении первой соответствующей параметрам конфигурации

статического (способного преобразовывать один входной вектор в требуемый выходной) и динамического (способного преобразовывать несколько входных векторов в требуемые выходные вектора) триггерного устройства в сравнении с прямым перебором. Для случая динамической структуры задавались два входных и выходных вектора, обработка осуществляется с использованием слоя межкомпонентных связей «Шина».

Таблица 1

Размер конфигурации ДО	2X2		3X3		4X4		5X5		6X6		7X7		8X8		9X9		10X10		
	Статическая (С), динамическая (Д)	С	Д	С	Д	С	Д	С	Д	С	Д	С	Д	С	Д	С	Д		
Среднее время поиска первой конфигурации удовлет. критериям прямым перебором, мс		23,41	38,12	74,62	631,12	25,17*10 ²	13,68*10 ³	81,34*10 ³	64,29*10 ⁴	23,48*10 ⁴	12,54*10 ⁵	43,92*10 ⁴	17,79*10 ⁵	50,19*10 ⁴	43,48*10 ⁵	74,87*10 ⁴	13,77*10 ⁶	14,11*10 ⁶	27,35*10 ⁶
Среднее время поиска первой конфигурации удовлет. критериям с помощью адаптированного ГА на основе ВСП, мс		64,62	120,4	93,91	581,43	15,28*10 ²	68,19*10 ²	49,24*10 ²	47,81*10 ⁴	68,23*10 ³	72,78*10 ⁴	81,49*10 ³	13,28*10 ⁵	92,56*10 ³	18,34*10 ⁵	10,3510 ⁴	37,14*10 ⁵	16,97*10 ⁵	52,31*10 ⁵

Из приведенных данных видно, что преимущество предложенного ГА на ВСП над методом прямого перебора начинается с конфигурирования ДО размерностью 3X3 в случае динамической структуры объекта и 4X4 при статической структуре.

В процессе исследования метода была выполнена проверка влияния мощности начальной популяции на время поиска решения.

На рисунке 10 отображен график зависимости времени, требуемого для нахождения конфигурации триггерного устройства, от начальной популяции, конфигурация имела прямоугольный вид размерностью 3X4 с перестраиваемой структурой, с использованием межкомпонентной шины в виде «вырожденных» сетей Петри с входным вектором 101010 и выходным вектором 011010.

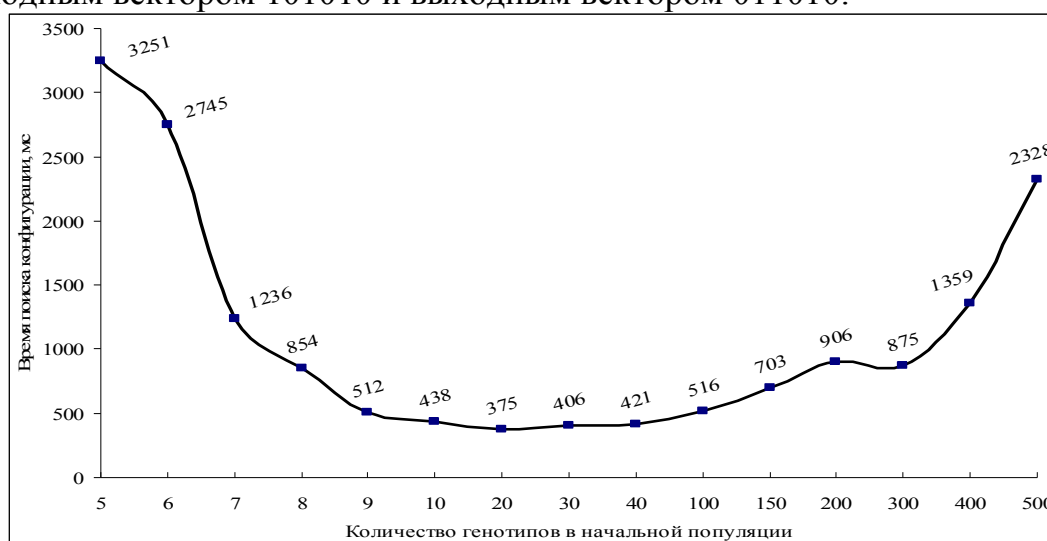


Рисунок 10. Среднее время поиска конфигурации в зависимости от размера начальной популяции, мс

Качественный показатель представленного графика в зависимости от размерности ДО не изменялся.

Основные результаты и выводы

При выполнении диссертационной работы были получены следующие результаты:

1. На основе проведенного анализа существующих математических моделей конфигурирования дискретных объектов был сделан вывод о необходимости создания имитационных моделей, учитывающих динамическое изменение связей в процессе функционирования объекта с заданным поведением на основе генетических алгоритмов и сетей Петри;

2. Реализована адаптация и формализация генетического алгоритма, с помощью вложенных сетей Петри для решения задачи конфигурирования ДО с заданным поведением;

3. Предложена математическая модель для формирования конфигурации ДО с фиксированной и перестраиваемой структурой;

4. Разработана архитектура информационной системы, основанной на полученных математических моделях;

5. Проведена оценка эффективности разработанных моделей на примере конкретных практических задач конфигурирования вычислительной техники с заданным поведением и формирования расписания занятий вуза на основе реализованных программных средств.

ОСНОВНЫЕ ПУБЛИКАЦИИ ПО ТЕМЕ ДИССЕРТАЦИИ

Статьи в научных изданиях, входящих в перечень рекомендованных ВАК

1. Петросов, Д.А. Адаптация генетического алгоритма при моделировании вычислительной техники с изменяющейся структурой и набором компонентов на основе Сетей Петри [Текст]/ **Д.А. Петросов** // Вопросы современной науки и практики. – Тамбов: Изд-во ТГТУ, 2009. – №6(20). – С. 54–63.

2. Петросов, Д.А. Математическое моделирование формирования конфигураций вычислительной техники с заданным поведением [Текст]/ **Д.А. Петросов** // Вопросы современной науки и практики. – Тамбов: Изд-во ТГТУ, 2009. – №7(21). – С. 113–121.

3. Петросов, Д.А. Математическая модель формирования конфигурации вычислительной техники на основе триггеров [Текст]/ **Д.А. Петросов** // Вестник ИжГТУ. – Ижевск: Изд-во ИжГТУ, 2009. – №3(43). – С. 139–143.

Статьи в научных журналах и сборниках трудов

4. Ельчанинов, Д.Б. Применение генетических алгоритмов при проектировании компьютерной техники [Текст]/ Д.Б. Ельчанинов, Механа Сами, **Д.А. Петросов** // Вестник Херсонского государственного университета. – Херсон: Изд-во ХГУ, 2003. – № 2(18). – С. 146–148.

5. Лобода, В.Г. Концепция построения структур функционально ориентированных вычислительных устройств [Текст]/ В.Г. Лобода, **Д.А. Петросов** // АСУ и приборы автоматики. – Харьков: Изд-во ХНУРЭ, 2003. – №122. – С. 61–71.

6. Белова, Н.В. Концепция буферизации в вычислительных устройствах и системах. [Текст]/ Н.В. Белова, Механа Сами Саади, **Д.А. Петросов** // Радиоэлектроника и молодежь в XXI веке: сб. научных трудов 6 – го Международного молодежного форума. – Харьков: Изд-во ХНУРЭ, 2002. – Ч.2. – С.36–37.

Статьи в материалах и сборниках трудов научных конференций

7. Ельчанинов, Д.Б. Представление генетических алгоритмов сетями Петри в задачах проектирования компьютерной техники [Текст]/ Д.Б. Ельчанинов, В.Г. Лобода, **Д.А. Петросов** // Информационные технологии – в науку и образование: сб. материалов научно-практической конференции. – Харьков: Изд-во ХНУРЭ, 2005. – С. 48–51.

8. Белова, Н.В. Модель перепрограммируемого процессора с гибкой архитектурой [Текст]/ Н.В. Белова, Т.Г. Долженкова, **Д.А. Петросов** // Радиоэлектроника и молодежь в XXI веке: сб. материалов 8-го Международного молодежного форума. – Харьков: Изд-во ХНУРЭ, 2004. – Ч.2. – С. 277.

9. Белова, Н.В., Минимальное замкнутое покрытие двух классов цифровых автоматов [Текст]/ Н.В. Белова, Т.Г. Долженкова, **Д.А. Петросов** // Радиоэлектроника и молодежь XXI веке: сб. материалов 7-го Международного молодежного форума. – Харьков: Изд-во ХНУРЭ, 2003. – С. 468.

10. **Петросов, Д.А.** Интеллектуальная система поддержки принятия решений при формировании вычислительной техники с использованием генетических алгоритмом и L-сетей Петри [Электронный ресурс]/ Д.А. Петросов// сб. материалов Международной научно-практической конференции студентов, аспирантов и молодых ученых БГТУ им. В.Г. Шухова. – Белгород, 2009. 1 электрон. опт. диск (CD-ROM).

11. **Петросов, Д.А.** Математическая модель процесса формирования конфигурации вычислительной техники на основе триггеров с использованием генетических алгоритмов и сетей Петри [Текст]/ Д.А. Петросов// Проблемы управления, передачи и обработки информации (АТМ-ТКИ 50): сб. тр. Междунар. науч. конф. – Саратов: Изд-во СГТУ, 2009. – С.344–347.

12. Иванов, И.В. Концепция работы межкомпонентной шины при конфигурировании дискретных объектов с заданным поведением [Текст]/ И.В. Иванов, **Д.А. Петросов** // Проблемы сельскохозяйственного производства на современном этапе и пути их решения: сб. материалов XIV международной научно-производственной конференции. – Белгород: Изд-во БелГСХА, 2010. – Ч.2. – С.23.

Официальная регистрация программ

13. Басавин, Д.А. Программа поиска конфигурации вычислительной техника/ Д.А. Басавин, **Д.А. Петросов**// Свидетельство о государственной регистрации программы для ЭВМ 2009616324 ФГУ ФИПС – 2009.

14. Карамышев, Е.П. Программа составления расписания учебного процесса ВУЗа/ Е.П. Карамышев, **Д.А. Петросов**// Свидетельство о государственной регистрации программы для ЭВМ № 2010612452 ФГУ ФИПС – 2010.

Подписано в печать 02.07.10.
Усл. печ. л. 1,00

Формат 60x84/16
Тираж 100 экз. Заказ №

Отпечатано в Белгородском государственном
университете им. В.Г. Шухова
308012, г. Белгород, ул. Костюкова, 46