

**Міжнародна наукова конференція**

**Інтелектуальні системи прийняття рішень та  
проблеми обчислювального інтелекту**

**ISDMCI'2011**

**Збірка наукових праць у двох томах**

**Том 2**

**Аналіз та моделювання складних систем і процесів  
Теоретичні і прикладні аспекти систем прийняття  
рішень**

**Обчислювальний інтелект та індуктивне моделювання  
Безпека інформаційних систем і мереж  
Теоретичні і прикладні аспекти удосконалення  
транспортних систем**

**Євпаторія – 2011**

К. С. Бовсупольська, К. Х. Зеленський МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ АЕРОДИНАМІЧНИХ ПРОЦЕСІВ У ЦИКЛОННИХ АППАРАТАХ .....	193
А. Б. Брюховецкий, В. Е. Савалевич, А. М. Кожухов, В. Н. Диков, А. Л. Поляков ОЦЕНКА БЛИЗКА АСТЕРОИДА ПО АМПЛИТУДЕ ЕГО СИГНАЛА НА ПЭС-КАДРЕ ПРИ ОБРАБОТКЕ ОПТИЧЕСКИХ НАБЛЮДЕНИЙ .....	195
Н. Б. Васильєва ФОНЕМНЕ РОЗПІЗНАВАННЯ ЗЛИТОГО МОВЛЕННЯ НА ОСНОВІ ВІЛЬНИХ ГРАМАТИК ФОНЕМНОГО ТА СКЛАДОВОГО РІВНЯ .....	196
К. В. Волосюк ЗАДАЧА ОПТИМАЛЬНОГО РОЗКРОЮ МАТЕРІАЛУ ПРИ ПРОЕКТУВАННІ МЕБЕЛІ .....	200
М. В. Волченко ОПТИМИЗАЦИЯ АЛГОРИТМА ВЫВОДА В ЛОГИКЕ ВЫСКАЗЫВАНИЙ ДЛЯ 0,1-НОГО ПРЕДСТАВЛЕНИЯ ДИЗЪЮНКТОВ .....	202
Е. В. Волченко ПОПОЛНЕНИЕ ВЗВЕШЕННЫХ ОБУЧАЮЩИХ ВЫБОРОК W-ОБЪЕКТОВ, ПОСТРОЕННЫХ НА ОСНОВЕ СЕТОЧНОГО ПОДХОДА В АДАПТИВНЫХ СИСТЕМАХ РАСПОЗНАВАНИЯ .....	205
А. С. Гавриленко ОПТИМИЗАЦИЯ АЛГОРИТМОВ СИНТЕЗА СИСТЕМ КЛАССИФИКАЦИИ СИГНАЛОВ .....	209
А. И. Гуда, А. И. Михалев ИДЕНТИФИКАЦИЯ СИСТЕМ С ХАОТИЧЕСКОЙ ДИНАМИКОЙ .....	211
С. В. Домль, Е. С. Козелкова, Д. П. Папенок МЕТОДИКА ПОВЫШЕНИЯ КАЧЕСТВА РАДИОЛОКАЦИОННОГО ИЗОБРАЖЕНИЯ КОСМИЧЕСКИХ СИСТЕМ ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ ЗЕМЛИ ДЛЯ РЕШЕНИЯ ЗАДАЧ КАРТОГРАФИИ .....	212
О. С. Загвадін, Ю. В. Крак, А. І. Куляс МЕТОД ВИЗНАЧЕННЯ ПОЗИЦІЇ ЗМІНИ ДИКТОРА У ГОЛОСОВИХ МОВНИХ СИГНАЛАХ .....	213
Д. Д. Зербіто, Ю. В. Цимбала, О. Я. Різник, Ю. Є. Князь, І. Ю. Юрчак ДО СТВОРЕННЯ СПЕЦІАЛІЗОВАНИХ МОВ РОЗПІЗНАВАННЯ .....	217
Г. В. Кіт СПЕКТРАЛЬНИЙ МЕТОД КЛАСИФІКАЦІЇ БІОМЕДИЧНИХ СИГНАЛІВ .....	218
А. М. Кожухов, В. Е. Савалевич, А. Б. Брюховецкий, В. П. Власенко, А. Н. Богдановский МЕТОД ОЦЕНКИ КООРДИНАТ АСТЕРОИДА НА ДИСКРЕТНОМ ИЗОБРАЖЕНИИ ПРИ ОБРАБОТКЕ ОПТИЧЕСКИХ НАБЛЮДЕНИЙ .....	219
А. Г. Кондріна, Є. А. Савченко, О. Е. Деректоренко ПРОГНОЗУВАННЯ ІНДЕКСУ РОЗВИТКУ ЛЮДСЬКОГО ПОТЕНЦІАЛУ УКРАЇНИ ЗА КОМБІНАТОРНИЙ АЛГОРИТМОМ МГУА .....	221
Л. О. Коршевнюк ВВЕДЕНИЯ ПРИХОВАННЫХ ВЕРШИН У СТРУКТУРЫ МЕРЕЖ БАЙЕСА .....	224
Ю. В. Крак, Б. А. Троцюк, Ю. В. Барчукова КЛАСИФІКАЦІЯ ДАКТИЛЕМ ЖЕСТОВИХ МОВ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ НАВЧАЛЬНОЇ СИСТЕМИ ДАКТИЛЬНОЇ АБЕТКИ ЖЕСТОВОЇ МОВИ ДЛЯ УКРАЇНСЬКОЇ ТА ПОЛЬСЬКОЇ ЖЕСТОВИХ МОВ .....	229
Ю. Г. Кривонос, Ю. В. Крак, А. С. Тернов, М. П. Івняк СТРУКТУРНО-ВІЗЕМНИЙ АНАЛІЗ ВІЗУАЛЬНОЇ СКЛАДОВОЇ МОВЛЕННЕВОГО ПРОЦЕСУ З ВИКОРИСТАННЯМ ТРИВИМІРНОЇ МОДЕЛІ ГОЛОВИ ЛЮДИНИ .....	231
А. І. Куляс, В. І. Сидоренко, М. М. Шатковський ВИБІР ОЗНАК ПРИРОДНОСТІ ЗВУЧАННЯ ДЛЯ ПРОЛЕМИ КОМП'ЮТЕРНОГО КОНКАТЕНАТИВНОГО СИНТЕЗУ УКРАЇНСЬКОГО МОВЛЕННЯ .....	232

Треба враховувати те, що є можливість обертати фігури довільним чином. Тому треба порівнювати саме площі фігур і пустих місць при зміні популяції. Отже, важливим кроком є саме формування початкової популяції.

Використовуючи введені означення, наведемо алгоритм розв'язку задачі:

1. Формування початкової популяції за допомогою декодера.
  - 1.1. Розташовуємо поточну фігуру на пластині, не порушуючи обмеження (2) – (6).
  - 1.2. Аналізуємо пусті місця. Якщо є фігура, площа якої максимально підходить до площі пустого місця, розташовуємо її на тому місці (допускається здійснення повороту фігури).
  - 1.3. Перевіряємо, чи заповнена пластинка повністю. Якщо так, переходимо до кроку 2, інакше – до кроку 1.1
2. Розрахувати пристосованість кожної особини згідно з (1).
3. Створюється нова особина за допомогою схрещування, мутації або селекції.
4. Повторюємо кроки 2-3 поки не виникне деяка умова закінчення пошуку.
5. В якості результату розв'язання задачі вибираємо ту особину, значення цільової функції (1) для якої є мінімальним.

**Висновки.** В даній роботі було запропонований метод розв'язання задачі оптимального розкрою матеріалу за допомогою генетичного алгоритму. Приведений алгоритм дає можливість оптимально розташувати всі фігури, які необхідні для виготовлення меблів. Це сприяє економічному використанню матеріалів, що має практичне значення для підприємств, які проектують та виготовляють меблі.

#### ЛІТЕРАТУРА

1. Попова А.В. Генетические алгоритмы на примерах решения задач раскроя // Проблемы управления. – 2008. – №2. – С. 57-63.
2. Томас Х. Кормен и др. NP-полнота // Алгоритмы: построение и анализ – 2-е изд. – М.: «Вильямс», 2006. – 1296 с.
3. Ермаченко А.И. Модели и методы решения задач прямоугольного раскроя и упаковки на базе метаэвристики "Поиск с запретами" [Электронный ресурс] / Режим доступа: <http://www.dissereat.com/content/>

### ОПТИМИЗАЦІЯ АЛГОРИТМА ВИВОДА В ЛОГІЦКЕ ВИСКАЗУВАННІЙ ДЛЯ 0,1-НОГО ПРЕДСТАВЛЕННЯ ДИЗ'ЮНКТОВ

М. В. Волченко

Донецький державний університет інформатики та штучного інтелекту, г. Донецьк,  
[volchovy4@gmail.com](mailto:volchovy4@gmail.com)

В настоящей работе рассматривается задача проверки противоречивости множества дизъюнктов, которая находит свое применение как инструмент формального доказательства в системах верификации программ, системах принятия решений, дедуктивных базах данных, информационно-поисковых системах [1].

Одной из основных проблем выполнения процедур дедуктивного вывода является экспоненциальный рост пространства поиска, что существенно увеличивает временные и ресурсные затраты на поиск решения. Для эффективной обработки множеств дизъюнктов большой мощности процедура вывода должна отвечать следующим требованиям [2]:

- насколько возможно, сужать пространство поиска контрарной пары на каждом шаге резольвирования;
- исключать дизъюнкты, которые не могут быть использованы в процессе доказательства;
- эффективный алгоритм выбора контрарных пар для резольвирования.

Одним из эффективных способов представления множества дизъюнктов логики высказываний является граф связей [2]. Процедура вывода на графе связей удовлетворяет всем представленным выше принципам создания эффективных процедур дедуктивного вывода, однако требует пересчета графа связей на каждом шаге резольвирования с вычислением новых связей. Также эта процедура требует генерации некоторого количества дополнительных дизъюнктов на промежуточных этапах и не предоставляет возможности оценки связи, выбираемой для резольвирования.

Идея представления дизъюнктов в виде 0-1-ных наборов, представление графа связей в виде треугольной матрицы, а также алгоритм резольвения для такого представления были предложены в [3]. Настоящая работа является ее развитием и состоит в оптимизации процедуры построения матрицы связей и ее обработки.

### 1 Постановка задачи

Рассматривается классическая задача проверки выполнимости логических высказываний. Исходными данными является множество дизъюнктов  $S = \{C_1, C_2, \dots, C_k\}$ . Необходимо проверить выводим ли из множества дизъюнктов  $S$  заданного высказывания пустой дизъюнкт  $\perp$ . Если это так, то  $S$  невыполнимо, иначе – выполнимо [2].

Данная задача является NP-полной за счет экспоненциального роста пространства поиска.

Одним из основных методов повышения эффективности процесса вывода является резольвирование на графе связей [2]. При построении графа связей для последовательности дизъюнктов  $S = C_1, C_2, \dots, C_k$ , состоящих из конъюнктов  $K_1, K_2, \dots, K_m$ , каждому дизъюнкту ставится в соответствие вершина графа  $G$ . Две вершины соединяются ребром, называемым связью, если они образуют контрадную пару, т.е. по этим дизъюнктам может быть проведено резольвирование. Результат резольвирования узлов добавляется в граф  $G$ .

Последовательный алгоритм вывода на графе связей имеет следующий вид [2]:

1. Выбор связи из множества связей.
2. Резольвирование связи и получение резольвенты. Удаление связи, по которой произошло резольвирование.
3. Если получена пустая резольвента, то успешное завершение, иначе помещение резольвенты в граф, добавление ее связей, удаление дизъюнктов-тавтологий и чистых дизъюнктов, выполнение операции поглощения дизъюнктов.
4. Если граф не содержит ни одного дизъюнкта, то неуспешное завершение алгоритма, иначе переход к шагу 1.

На основе этого алгоритма, с целью сокращения объемов выполняемых вычислений автором в [3] был предложен алгоритм резолюции для 0-1-ного представления дизъюнктов. В настоящей работе рассматривается способ упорядочивания 0-1-ных наборов при построении матрицы связей и соответствующие такому представлению эвристики выбора связи для резольвирования и добавления резольвенты в матрицу связей.

### 2 Представление дизъюнктов и их связей

Пусть задано множество дизъюнктов  $S$ . Пусть  $A = \{a_1, a_2, \dots, a_n\}$  – алфавит всех букв из  $S$ . Считаем, что  $A$  линейно упорядочен.

На начальном этапе каждый дизъюнкт будем представлять 0,1-ным набором  $X_i = (X_{i1}, \dots, X_{in})$  по правилу [4]:

$$X_i = \begin{cases} 1, & \text{если } a \text{ входит в } C_i \text{ без отрицания} \\ 0, & \text{если } a \text{ входит в } C_i \text{ с отрицанием} \\ \_ & \text{если литерал } a \text{ не входит в } C_i \end{cases}$$

В результате такого представления будет получено множество 0,1-ных векторов (наборов), соответствующее исходному множеству  $S$ .

Полученные наборы  $X_i$  необходимо упорядочить таким образом, чтобы упростить поиск связей для резольвирования, а также избежать роста пространства поиска при добавлении резольвенты. В настоящей работе предлагается разбить наборы  $X_i$  на классы, содержащих соответствующее номеру класса число  $l_i$ , что соответствует длине дизъюнктов множества  $S$ . Такие классы необходимо упорядочить по убыванию числа  $l_i$  в наборах (возрастанию длины дизъюнктов множества  $S$ ). Внутри каждого из классов также необходимо упорядочить наборы  $X_i$  по возрастанию номера позиции  $i$ .

Каждой паре наборов  $X_i$  и  $X_j$  из списка дизъюнктов, ставится в соответствие связь  $R(i, j)$  по правилу:

$$R_m(i, j) = \begin{cases} 1, & \text{если } (X_{im} = X_{jm}) \text{ и } (X_{im} \neq \_) \\ 0, & \text{если } X_{im} \odot X_{jm} = 1 \\ \_ & \text{если } (X_{im} = \_) \text{ или } (X_{jm} = \_) \end{cases}$$

Далее по построенной матрице выполняется процедура резольвирования по следующему алгоритму.

### 3 Описание алгоритма резолюции на матрице связей

Предлагаемый алгоритм состоит из следующих процедур: преобразование матрицы связей, выбор связи для резольвирования, резольвирование дизъюнктов, добавление полученного вектора-резольвенты в матрицу связей.

### 3.1 Оптимизация матрицы связей.

Обработка матрицы осуществляется построчно, начиная с первой строки. Выполняется удаление дизъюнктов-тавтологий, поглощенных и чистых дизъюнктов для всех связей матрицы  $R$ .

1) В матрице связей, если для наборов  $X_i$  и  $X_j$  количество нулей в связи  $R(i,j)$  больше одного, то в результате резольвирования соответствующей пары дизъюнктов будет получен дизъюнкт-тавтология, и такую связь необходимо удалить.

2) Если  $\forall i: i \in \{1, n\} (R_i(i, j) = R_i(j, j)) \cup (R_i(i, j) = \_)$ , то дизъюнкт  $X_i$  поглощает дизъюнкт  $X_j$ , удаляются  $j$ -я строка и  $i$ -я строка.

3) Если в текущей строке  $j$  отсутствуют связи для резольвирования, то получен чистый дизъюнкт и эта строка удаляется.

Обработка матрицы связей продолжается до тех пор, пока не будут рассмотрены все строки.

Так как поглощающие дизъюнкты имеют меньшую длину, чем поглощаемые, то упорядочивание наборов по числу  $\_$  позволяет выполнить процедуру поглощения за меньшее число шагов.

### 3.2 Выбор связи для резольвирования.

Для получения резольвенты минимальной длины в [4] предложена эвристика выбора связи для резольвирования: выбирается связь, число  $\_$  которой минимально отличается от числа  $\_$  второго дизъюнкта и число единиц максимально. В большинстве случаев такой подход позволяет избежать резкого роста числа обрабатываемых дизъюнктов.

Благодаря способу упорядочивания наборов и построения матрицы связей, предложенных в данной работе, выбор связи с помощью такой эвристики сводится к последовательному выбору связей для резольвирования, что значительно уменьшает число, выполняемых сравнений.

В результате исследований, проведенных на ряде тестовых примеров, было показано, что такое упорядочивание наборов позволяет анализировать матрицу связей, определять пары наборов, резольвирование которых приведет к поглощению этих наборов. Если  $\forall i: i \in \{1, n\} (R_i(i, j) = R_i(j, j)) \cup (R_i(i, j) \oplus R_i(j, j) = 1)$ , то наборы  $X_i$  и  $X_j$  будут удалены.

Так как выбор таких связей приводит к сокращению пространства поиска, предлагается использовать такое свойство на этапе оптимизации матрицы связей, а также при добавлении резольвенты в матрицу связей. Следует отметить, что такую проверку следует проводить для наборов одного класса, содержащих одинаковое число  $\_$ , таким образом, использование такой эвристики существенно не повышает вычислительную сложность алгоритма.

### 3.3 Операции резольвирования дизъюнктов.

Наличие в 0,1-ном наборе связи одного нуля показывает, что дизъюнкты образуют контрарную пару, то есть возможно их резольвирование. Операция резольвирования заключается в получении из векторов  $X_i$  и  $X_j$  нового вектора-резольвенты  $T_{ij}$ , который определяется по правилу:

$$T_{ij} = \begin{cases} X_i, & \text{если } (X_i \neq X_j) \text{ и } (X_i \neq \_) \\ X_j, & \text{если } (X_i \neq X_j) \text{ и } (X_j \neq \_) \\ X_i, & \text{иначе} \end{cases}$$

Полученные резольвенты добавляются в соответствующий класс наборов, не нарушая упорядоченности внутри этого класса, что позволит уменьшить количество сравнений при поиске новой связи для резольвирования, а также увеличить вероятность поглощения новым дизъюнктом предыдущих.

Если номер класса нового набора меньше или равен номерам классов наборов-«родителей», то необходимо проверить возможность поглощения ими других наборов дизъюнктов и провести операцию поглощения. Для добавляемой резольвенты необходимо определить связи с другими дизъюнктами, которые набор-резольвента наследует у дизъюнктов-«родителей». В результате резольвирования новых связей не возникает.

Если на некотором этапе выполнения алгоритма получена пустая резольвента, то S-выполнимо и алгоритм заканчивает свою работу. Если на некотором этапе выполнения алгоритма невозможно найти пару дизъюнктов для резольвирования, то S- невыполнимо.

### Выводы

В настоящей работе предложено новый способ упорядочивания 0-1-ных наборов при построении матрицы связей и соответствующим такому представлению эвристики выбора связи для резольвирования и добавления резольвенты в матрицу связей, а также выделена эвристика, позволяющая определять связи дизъюнктов, которые будут поглощены в результате резольвирования.

В результате тестирования алгоритма на ряде тестовых примеров было получено, что число выполняемых итераций, т. е. время выполнения алгоритма, уменьшается в среднем на 8%, а рост пространства поиска по сравнению с выводом по графе связей в среднем уменьшается на 18%, и составляет – 8%.

#### СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. C.P.Gomez, H.Kautz, A. Sabharwal and B.Selman. Satisfiability Solvers// Handbook of Knowledge Representation, 2008.
2. Вагин В.Н., Головина Е.Ю., Загорянская А.А., Фомина М.В. Достоверный и правдоподобный вывод в интеллектуальных системах / Под ред. В.Н. Вагина, Д.А. Певислова. – М.: ФИЗМАТЛИТ, 2004.
3. Волченко М.В. Алгоритм резолюции в логике высказываний при 0-1-ном представлении цизъюнктов / И.С. Грунский, М.В. Волченко // XII национальная конференция по искусственному интеллекту «КИИ-2010». – М.: Физматлит, 2010. – Т.1. – С.64-71.
4. Волченко М.В. Выбор эвристик для алгоритма резолюции / М.В. Волченко // Материалы IV Международной научно-практической конференции «Сучасна інформаційна Україна: інформатика, економіка, філософія». – Донецьк, 2010. – Т.1. – С. 94-91.

#### ПОПОЛНЕНИЕ ВЗВЕШЕННЫХ ОБУЧАЮЩИХ ВЫБОРОК W-ОБЪЕКТОВ, ПОСТРОЕННЫХ НА ОСНОВЕ СЕТОЧНОГО ПОДХОДА В АДАПТИВНЫХ СИСТЕМАХ РАСПОЗНАВАНИЯ

Е. В. Волченко

Государственный университет информатики и искусственного интеллекта МОН Украины,  
г. Донецк [lviv@mail.promtele.com](mailto:lviv@mail.promtele.com)

Динамичность изменения объектов и процессов, происходящих в окружающем мире, требует разработки подходов, позволяющих адаптировать автоматические системы к таким изменениям. Данная проблема является одной из наиболее актуальных в современной теории построения систем автоматического распознавания [1]. Наравне с требованием эффективности классификации, современные системы распознавания должны отвечать требованиям адаптивности и работы в режиме реального времени [2, 3]. Под адаптивностью принято понимать [2] способность системы изменять свои свойства (словарь признаков, обучающую выборку, решающие правила классификации и т.д.) в соответствии с изменениями распознаваемых объектов. В обучающихся системах распознавания изменения распознаваемых объектов отражаются в новых объектах, добавляемых в обучающую выборку в процессе работы системы. Работа в реальном времени требует построения решающих правил, позволяющих принять решение о классификации за выделенное время [3].

Одним из способов обеспечения адаптивности и быстродействия современных систем распознавания является предобработка обучающих выборок, которая включает в себя очистку, сжатие и объединение данных [2]. Для адаптивных систем распознавания, на наш взгляд, наиболее важным является решение проблемы сжатия данных, поскольку постоянное, для некоторых систем достаточно интенсивное, пополнение обучающих выборок может привести к существенному увеличению времени на построение решающих правил и выполнение классификации, а также снижению эффективности работы систем за счет переобучения.

Задача сжатия данных при условии неизменяющегося словаря признаков может быть решена путем отбора некоторого множества объектов исходной обучающей выборки, каждый из которых отвечает предъявляемым требованиям, или построением множества новых объектов, каждый из которых строится по информации о некотором подмножестве объектов исходной обучающей выборки и обобщает его.

Алгоритмы, реализующие отбор некоторого множества объектов исходной обучающей выборки, получили название «близоруких», поскольку выбор некоторого множества объектов не учитывает расположения всех объектов исходной выборки в пространстве признаков, а невыбранные в это множество объекты удаляются. К такого типа алгоритмам относятся алгоритмы STOLP [4], FRIS-STOLP [5], NNDE (Nearest Neighbor Density Estimate) и MDCA (Multiscale Data Condensation Algorithm) [1]. Различаются эти алгоритмы способом отбора объектов, используемой мерой вычисления расстояния между ними и критерием оптимальности полученного сокращенного множества объектов.

В основе алгоритмов, выполняющих построение новой выборки на основе обобщенной информации об объектах исходной выборки, лежит дискретизация пространства признаков и анализ