

УДК 681.31

Ю.О. Чернышев, Н.Н. Венцов

**ИССЛЕДОВАНИЕ ВАРИАНТОВ ОПТИМИЗАЦИИ  
ИНФОРМАЦИОННОГО ОБЕСПЕЧЕНИЯ САПР СБИС, В ТОМ ЧИСЛЕ И  
НАНОМЕТРОВОЙ ГЕОМЕТРИИ, ПРИ ПОМОЩИ МЕТОДОВ  
ЭВОЛЮЦИОННОЙ И ПАРАМЕТРИЧЕСКОЙ АДАПТАЦИИ**

**Введение.** Автоматизация проектирования – это многоаспектная и многоуровневая научная задача. Особенностью проектирования СБИС является проблема «проклятия размерности», сущность которой заключается не только в большой области поиска решения, но и в необходимости обработки огромных массивов информации описывающих объект проектирования [1]. В настоящее время актуальной научной задачей является разработка САПР СБИС, ориентированных на работу в нанометровых проектах. При построении САПР такого класса одной из острых проблем является взрывной характер роста проектных данных [2]. При построении эффективных механизмов доступа к локальным информационным массивам проектных данных, большого объема и сложной структуры, ключевое значение имеет сокращение числа операций дискового ввода-вывода [3].

Для улучшения компонент информационного обеспечения средств проектирования СБИС нанометровой геометрии, разрабатываемых компанией Mentor Graphics (рис. 1), в [2] предлагается использовать сервисы аналитической обработки (OLAP-системы).



Рис. 1. Организация цикла проектирования электронного устройства и внедрение экспертной системы

Заявленное время обработки запросов OLAP-системами составляет около 0,1% от времени выполнения подобных запросов в хранилище [2]. OLAP-система делает мгновенный снимок хранилища данных и преобразует его в пространственную модель (модель куба данных), оптимизированную для выполнения запросов [2]. Одной из составляющих OLAP-систем являются многомерные базы данных [4, 5]. Недостатком многомерных баз данных является неэффективное использование памяти, которая заранее резервируется для всех значений даже, если часть из них заведомо будет отсутствовать [4, 5].

Многомерная модель куба данных это логическое представление информации, физически на диске данные хранятся в виде некоторой структуры [5]. Поэтому при построении хранилищ данных основой может служить реляционная модель хранения данных. При таком подходе гиперкуб эмулируется системой управления

базами данных (СУБД) на логическом уровне. СУБД данного типа могут хранить значительно большие объемы данных, хотя проигрывают реальным многомерным БД по скорости выполнения запросов [4, 5].

**1. Формулировка проблемы.** Из выше сказанного следует, что основу информационного обеспечения САПР, ориентированных на разработку СБИС нанометрового уровня, могут составить базы данных построенные на основе реляционного подхода. При такой организации банка данных наиболее ресурсоемкой является операция соединения отношений, входящих в состав БД САПР СБИС. В связи с чем, выбор оптимального порядка соединения отношений является ключевой проблемой организации информационного обеспечения САПР СБИС. В настоящее время для решения задачи выбора оптимального порядка соединения отношений, расположенных на одном узле САПР, используются модификации метода ветвей и границ, методы поиска в локальной окрестности. Но наибольшее распространение получили динамическое программирование и жадный алгоритм. Недостатком метода динамического программирования, гарантирующего нахождение оптимального решения, является экспоненциальная временная сложность. По этой причине наиболее часто используется жадный алгоритм, обладающий меньшей вычислительной сложностью, но не гарантирующий нахождение глобального оптимума [3]. Авторами настоящей статьи установлено, что абсолютное отклонение решений, полученных при помощи жадного алгоритма, может быть весьма значительным (60-100 %) и не зависит от размерности задачи.

Известно, что перспективными методами решения оптимизационных задач являются генетические алгоритмы (эволюционная адаптация) и алгоритмы случайного поиска (параметрическая адаптация) [1, 6-8].

**2. Постановка задачи.** Задача выбора оптимального порядка соединения отношений, расположенных на одном узле САПР, формулируется следующим образом [3]. Имеется множество отношений  $R = \{r_0, r_1, r_2, \dots, r_i, \dots, r_{n-1}\}$ . Мощность отношения, полученного в результате соединения двух отношений  $r_i$  и  $r_j$ , имеющих общий атрибут соединения, определяется по формуле:

$$T(r_i \diamond r_j) = \frac{T(r_i) \cdot T(r_j)}{\max(V(r_i.x1), V(r_j.y1))}, \quad i, j \in \{0, 1, 2, \dots, n-1\}. \quad (1)$$

При наличии двух атрибутов соединения:

$$T(r_i \diamond r_j) = \frac{T(r_i) \cdot T(r_j)}{\max(V(r_i.x1), V(r_j.y1)) \cdot \max(V(r_i.x2), V(r_j.y2))}, \quad i, j \in \{0, 1, 2, \dots, n-1\} \quad (2)$$

в случае, если соединяемые отношения не имеют общего атрибута, мощность результирующего отношения определяется как произведение:

$$T(r_i \diamond r_j) = T(r_i) \cdot T(r_j), \quad i, j \in \{0, 1, 2, \dots, n-1\}, \quad (3)$$

где  $T(r_i)$  – количество кортежей в отношении  $r_i$ ;  $T(r_i \diamond r_j)$  – количество кортежей в отношении, являющемся результатом соединения отношений  $r_i$  и  $r_j$ ;  $V(r_i.x)$  – количество различных значений атрибута  $x$  отношения  $r_i$ ;  $x1, x2, y1, y2$  – атрибуты отношений по которым производится соединение.

Необходимо найти такой порядок соединения отношений  $\{r_0, r_1, r_2, \dots, r_i, \dots, r_{n-1}\}$ , чтобы суммарная мощность промежуточных отношений была минимальной.

Другими словами, требуется соединить отношения  $\{r_0, r_1, r_2, \dots, r_i, \dots, r_{n-1}\}$  в таком порядке, чтобы выполнялось условие:

$$\sum_{i=0}^{n-1} T_i(r^{(i)} \diamond R_\diamond[i]) \rightarrow \min, \quad (4)$$

где  $R_\diamond[i] = r^{(0)} \diamond r^{(1)} \diamond r^{(2)} \diamond \dots \diamond r^{(i-1)}$ .

**3. Предлагаемые подходы.** Для решения задачи выбора оптимального порядка соединения отношений, расположенных на одном узле САПР, на основе статического подхода к оптимизации запросов авторами разработаны генетический алгоритм (ГА) и алгоритмы случайного поиска. Гибкость структуры генетических алгоритмов, возможность ее настройки и перенастройки позволяют получить такую структуру, которая обеспечивает более высокий результат по сравнению с другими методами. Статический подход применяется в ситуациях, когда статистические характеристики базы данных с достаточной степенью достоверности описывают ее фактические параметры (мощности соединяемых отношений, распределение значений атрибутов и т.д.). Алгоритмы работают с решением, представленным в виде кортежа, содержащего идентификаторы соединяемых отношений. Исследования показали, что генетические алгоритмы, построенные на основе стандартных генетических операторов, обеспечивают достаточно низкую вероятность нахождения как оптимальных (0,01-0,02), так и субоптимальных (0,02-0,03) решений. В связи с чем, разработаны генетические операторы и операторы случайного шага, учитывающие особенности решаемой задачи и обеспечивающие отсечку бесперспективных решений. На основе предложенных генетических операторов формирования стартовой популяции, мутации ( $OM_i$ ), кроссинговера ( $OK_i$ ) и селекции ( $OS_i$ ) был разработан ГА. В качестве структуры поиска принят параллельно-последовательный подход. На первом этапе генерируется стартовая популяция. Формируемые генетическим алгоритмом стартовые популяции, как правило, не содержат оптимальные (приемлемые субоптимальные) решения. Для улучшения имеющихся решений используются модифицированные генетические операторы. Особи стартовой популяции разбиваются на заданное число групп (в зависимости от размерности задачи). Критерием разбиения является значение целевой функции для соответствующей особи. На рис. 2 представлен пример структуры генетического алгоритма, когда особи стартовой популяции разбиваются на четыре группы. В первую группу попадают особи с наименьшими значениями целевой функции, а в последнюю – с наибольшими. Далее внутри каждой группы применяются специфические операторы мутации, кроссинговера и селекции.

На основе предложенных операторов случайного шага разработаны: алгоритм случайного поиска с нелинейной тактикой и релаксационный алгоритм случайного поиска с линейной тактикой. Начальная альтернатива определяется алгоритмами при помощи процедуры формирования стартовой популяции разработанного генетического алгоритма.

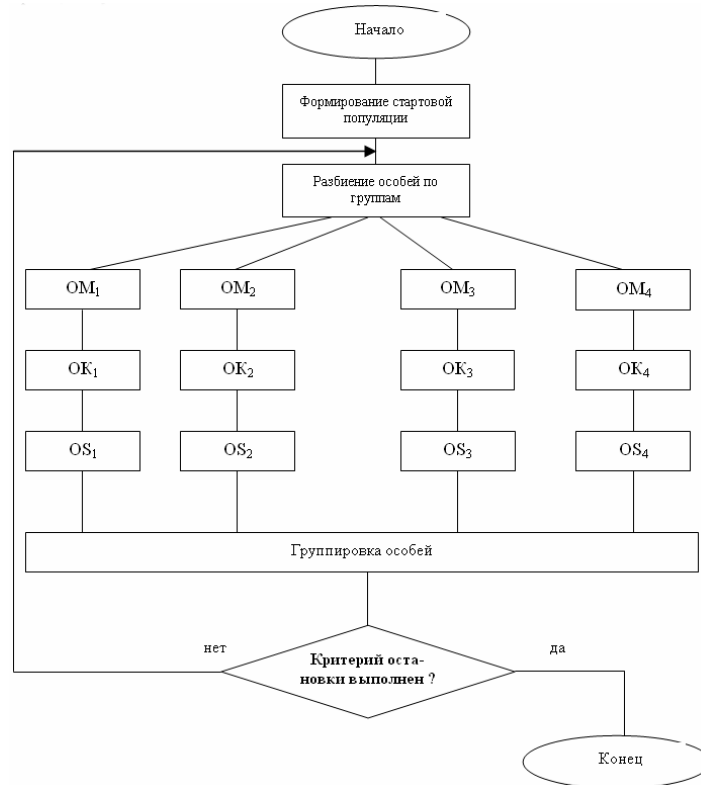


Рис. 2. Структурная схема генетического алгоритма

**Выводы.** Для определения сравнительной оценки вычислительной сложности разработанных алгоритмов с методом динамического программирования и жадным алгоритмом проведена серия экспериментов. Полученные результаты позволяют заключить:

- ◆ При решении задачи выбора оптимального порядка соединения отношений, расположенных на одном узле САПР, в соответствии со статическим подходом к оптимизации запросов, вероятность определения оптимального решения генетическим алгоритмом составляет 0,5-0,6, а субоптимального (отличающегося от оптимального не более, чем на 15%) – 0,7-0,85. На задачах размерности (количество соединяемых отношений) 8-15 экономия по времени составляет 15-20%, а на задачах размерности 15-30 – 25-30% по сравнению с методом динамического программирования.
- ◆ Разработанные алгоритмы случайного поиска позволяют определять приемлемые субоптимальные решения с вероятностью 0,4-0,5. Несмотря на то, что разработанным алгоритмам требуется на 20-30% больше времени по сравнению с жадным алгоритмом, вероятность улучшения решения на 8-12% составляет 0,7-0,9.

#### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Курейчик В.М., Лебедев Б.К., Лебедев О.Б. Поисковая адаптация: теория и практика. – М.: ФИЗМАТЛИТ, 2006.
2. Далекин А.В., Янушко В.В. Развитие средств автоматизированного проектирования // Труды международных научно-технических конференций «Интеллектуальные системы»

- (AIS'08) и «Интеллектуальные САПР» (CAD-2008). Научное издание в 4-х томах. – М.: Физматлит, 2008, Т.1. – С. 197-202.
3. *Гарсиа-Молина, Гектор, Ульман, Джефффри Д. Уидом, Джениффер.* Системы баз данных. Полный курс // Пер.с англ. – М.: Издательский дом “Вильямс”, 2003.
  4. Базы данных: Интеллектуальная обработка информации / В.В. Корнеев, А.Ф. Гареев, С.В. Васютин, В.В. Райх. – 2-е изд. – М.: Изд-во Молгачева С.В., 2001.
  5. *Архипенков С.Я.* Аналитическая система на базе Oracle Express OLAP: Проектирование, создание, сопровождение. – М.: Диалог-МИФИ, 1999.
  6. *Херрере Ф., Лозано М.* Нечеткие адаптивные генетические алгоритмы: дизайн, таксономии, а также будущие направления // Компьютерные программы. 7(2003), Спрингер - Верлаг, 2003. – С. 545-562.
  7. Руководство по генетическим алгоритмам. Выпуск I. Вашингтон, США, 1999.
  8. *Курейчик В.М., Лебедев Б.К., Лебедев О.Б., Чернышев Ю.О.* Адаптация на основе самообучения. Монография. – Ростов-на-Дону: Изд-во РГАСХМ ГОУ, 2004.

УДК 621.03: 007.5

**В.Б. Тарасов, С.В. Смагин**

### **ДИАЛОГИКА – ТЕОРЕТИЧЕСКАЯ ОСНОВА ИЗУЧЕНИЯ ВЗАИМОДЕЙСТВИЯ АГЕНТОВ В ИНТЕЛЛЕКТУАЛЬНЫХ САПР НОВЫХ ПОКОЛЕНИЙ\***

**Введение.** В начале XXI-го века все более актуальной становится разработка интеллектуальных систем автоматизированного проектирования (ИСАПР) новых поколений, призванных обеспечить существенный подъем эффективности процессов проектирования на всех стадиях разработки, значительное сокращение сроков проектирования, улучшение качества создаваемых искусственных (технических) систем. В таких ИСАПР должны быть реализованы инновационные стратегии проектирования, обеспечивающие повышение конкурентоспособности проектных организаций благодаря компьютерной интеграции ресурсов, расположенных в различных местах, интенсификации взаимодействия между партнерами, участвующими в процессах проектирования, обеспечению гибкости, надежности и своевременности принимаемых решений. Известными примерами подобных стратегий служат «совмещенное или параллельное проектирование» (Concurrent Design или Simultaneous Design), «кооперативное проектирование» (Cooperative Design или Collaborative Design), «эволюционное проектирование», «синергетическое эволюционное проектирование» (см., в частности, [1-3]).

В настоящей работе основное внимание уделяется теоретическим вопросам совершенствования подходов кооперативного проектирования, в особенности, проблемам «проектирования для некоторого внешнего или внутреннего заказчика», определяемого конкретной стадией жизненного цикла продукции (Design for X) [4], например, «проектирование для производства» (Design for Manufacturing), «проектирование для сборки» (Design for Assembly), «проектирование для логистики» (Design for Logistics) и т.п. Становление подобных «клиентоцентрических» подходов требует широкого внедрения в проектных организациях систем поддержки отношений с заказчиками CRM (Customer Relationship Management), кото-

---

\* Работа выполнена при поддержке РФФИ, гранты № 07-01-00656 и №07-07-00418.