

Полуян Анна Юрьевна

Ростовская-на-Дону государственная академия сельскохозяйственного машиностроения (РГАСХМ).

E-mail: pmivt@rgashm.ru.

344023, г. Ростов-на-Дону, ул. Страны Советов, 1.

Тел.: 8(863)258-91-36.

Кафедра прикладной математики и вычислительной техники; старший преподаватель.

Chernishev Yriy Olegovich

Rostov-on-Don State Agricultural Engineering Academy.

E-mail: pmivt@rgashm.ru.

1, Strani Sovetov street, Rostov-on-Don, 344023, Russia.

Phone: 8(863)258-91-36.

Applied Mathematics and Computer Facilities.

Head chair; professor.

Poluyan Anna Urievna

Rostov-on-Don State Agricultural Engineering Academy.

E-mail: pmivt@rgashm.ru.

1, Strana Sovetov Street, Rostov-on-Don, 344023, Russia.

Phone: 8(863)258-91-36.

Applied Mathematics and Computer Facilities; senior teacher.

УДК 519.7:004.8 + 004.8.023; 004.81.85

С.И. Родзин

**ОРГАНИЗАЦИЯ ПАРАЛЛЕЛЬНЫХ ЭВОЛЮЦИОННЫХ ВЫЧИСЛЕНИЙ
ПРИ ПОИСКЕ И ОПТИМИЗАЦИИ ПРОЕКТНЫХ РЕШЕНИЙ***

В статье рассматривается базовый цикл алгоритмов эволюционных вычислений. Предлагаются различные способы организации параллелизма: глобальный, миграционный, диффузионный. Приведены рекомендации по распараллеливанию операторов базового цикла эволюционных вычислений.

Эволюционные вычисления; параллелизм; репродукция; селекция.

S.I. Rodzin

**PARALLEL EVOLUTIONARY CALCULATIONS FOR THE SEARCH AND
OPTIMIZATION DESIGN APPROACH**

This article discusses the basic cycle of evolutionary algorithms for computing. Different ways of parallelism: a global, migration, diffusion. We give recommendations on the parallelization of operators basic cycle of evolutionary computation.

Evolutionary computation; parallelism; reproduction; selection.

Введение. Невозможно отрицать роль и значение параллельных вычислений и распределенных ресурсов в недетерминированных моделях интеллектуальных систем. Интеллектуальные САПР являются разновидностью интеллектуальных систем. За счет массового параллелизма интеллектуальные САПР могут обеспечить сокращение сроков проектирования, повысить аппаратно-программную на-

* Работа выполнена при поддержке: РФФИ (грант № 09-07-00318), г/б № 2.1.2.1652.

дежность системы, уменьшить объем передаваемых данных, повысить гибкость системы, динамичность решения задач проектирования.

Анализ алгоритмов и моделей, используемых в интеллектуальных САПР, свидетельствует о том, что возможности параллелизма существуют на различных стадиях и этапах проектирования и поиска когнитивных проектных решений [1]. Причем, параллелизм проявляется как на уровне представления проектных знаний и решений, так и на уровнях их поиска и обработки. Именно в этих областях особенно заметно влияние вычислительной трудоемкости, поскольку компьютеры с неймановской архитектурой не вполне подходят для решения интеллектуальных задач синтеза проектных решений; они предназначены, главным образом, для последовательной, детерминированной и числовой обработки информации. Трудности возникают как на уровне представления знаний, так и на уровнях управления, компиляции и выполнения проектных процедур.

Общепризнанными являются следующие характеристики интеллектуальных САПР:

- ◆ возможность обработки символьной информации;
- ◆ недетерминированная последовательность состояний системы;
- ◆ динамическая память и топология связей в системе;
- ◆ открытая архитектура;
- ◆ возможности организации параллельных вычислений, распределенного представления и управления знаниями.

Исследуем подробнее возможности организации параллельных вычислений и параллельной обработки проектных решений для широко применяемых в интеллектуальных САПР эволюционных моделей и алгоритмов.

Базовый цикл алгоритмов эволюционных вычислений. Эффективным средством решения многих проектных задач синтеза и оптимизации являются алгоритмы и модели эволюционных вычислений: генетические алгоритмы, генетическое программирование, эволюционные стратегии, эволюционное программирование, модели поведение роя пчел, стаи птиц, колонии муравьев, модели отжига, потока и другие конкурирующие эвристические алгоритмы [2]. Эволюционные вычисления широко применяются в интеллектуальных САПР.

Термин «эволюционные вычисления» (ЭВ) в интеллектуальных САПР подразумевает использование для решения разнообразных задач проектирования и оптимизации совокупности алгоритмических, программных, аппаратных средств и приближенных эвристических методов, инспирированных природными системами, для синтеза структур обработки проектных данных, а также на статистическом подходе к исследованию ситуаций и итерационном приближении к искомому проектному решению. Модели ЭВ являются инструментом исследования, способным конкурировать с другими методами поиска оптимальных решений по качеству получаемых решений в сочетании с трудоемкостью их получения.

Основным элементом системы ЭВ является базовый цикл алгоритма ЭВ. На рис. 1 приводится графическое изображение базового цикла алгоритма ЭВ.

Из рис. 1. видно, что базовый цикл включает следующую последовательность шагов: вычисление целевой функции, оценку качества решений, селективный отбор решений и репродукцию, т.е. создание новых решений. Структура базового цикла ЭВ обладает параллелизмом. Распараллеливание возможно как на уровне организации работы алгоритма, так и на уровне его компьютерной реализации.

Параллелизм на уровне компьютерной реализации означает вычисление на параллельных системах или процессорах значений целевой функции для разных решений, параллельное выполнение операторов ЭВ, что пропорционально повышает скорость работы алгоритма.

На уровне организации работы алгоритма ЭВ параллелизм может достигаться структурированием популяции решений.

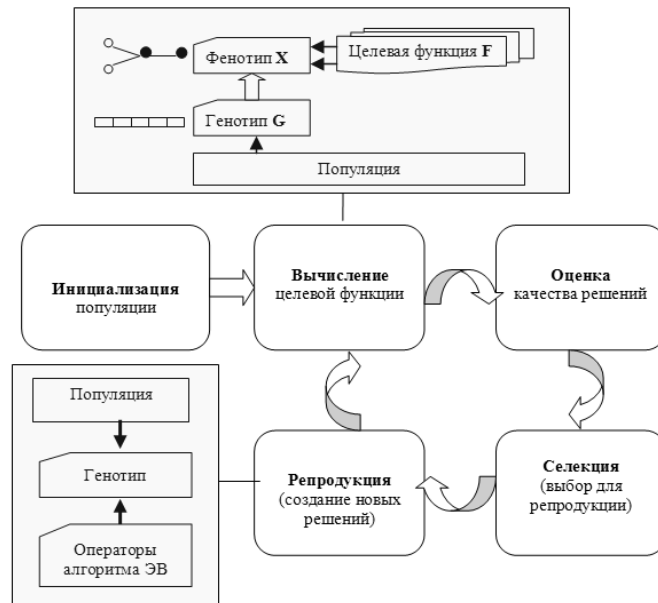


Рис. 1. Базовый цикл алгоритма эволюционных вычислений

Способы организации параллелизма. Оба подхода могут эффективно реализоваться на параллельных вычислителях, причем способы организации параллелизма могут быть различными: глобальный параллелизм, миграционная модель, диффузионная модель [3].

В случае глобального параллелизма речь идет о параллельном вычислении целевой функции и параллельном выполнении операторов алгоритма ЭВ. Целевая функция отдельного индивидуума в популяции, как правило, не зависит от других индивидуумов, поэтому вычисление целевых функций допускает распараллеливание. При этом популяция хранится в общей памяти, а отдельный процессор считывает хромосому из памяти и возвращает результат после вычисления целевой функции. Процедура синхронизации здесь проводится лишь при формировании новой популяции.

Если вычислительная система имеет распределенную память, то популяция и значения функции качества отдельных индивидуумов хранятся в «мастер»-процессоре, а расчеты производятся на вспомогательных процессорах. Некоторую трудность для распараллеливания представляет процедура селекции. Здесь преимущество в смысле распараллеливания получают те формы селекции, которые не используют глобальную статистику о популяции, например «соревновательные» способы селекции. Легко распараллеливаемыми являются операторы мутации и кроссинговера, так как они выполняются независимо друг от друга и в основном случайным образом.

Миграционный параллелизм наиболее эффективно можно организовать в вычислительных системах с MIMD-архитектурой путем разделения общей популяции на отдельные подпопуляции, эволюционное моделирование которых осуществляется на отдельных процессорах. Если между отдельными подпопуляциями установить правила миграции отдельных индивидуумов, то получится миграционная модель. Для построения миграционной модели требуется задать количество и размеры подпопуляций, топологию связей между ними, частоту и интервал миграции, а также стратегию эмиграции и иммиграции. Размеры и число подпопуляций определяются числом и быстродействием используемых процессоров или транспьютеров, а также сложностью решаемой проблемы. Что касается топологии связей в миграционной модели, то здесь различаются два варианта: «островная» модель, предусматривающая миграцию между любыми подпопуляциями, и модель, предусматривающая обмен индивидуумами между соседними подпопуляциями. Применение того или иного варианта миграционной модели зависит от архитектуры параллельного вычислителя (сетевая или кольцевая). Частота миграции определяется размерами подпопуляций. При высокой частоте подпопуляция может быстро потерять гетерогенность. Наоборот, при слабой интенсивности миграции и большом числе подпопуляций возрастает вероятность получения нескольких хороших решений, однако сходимость алгоритма замедляется.

Диффузионный параллелизм эффективно реализуется в вычислительных системах с SIMD-архитектурой. В этом случае общая популяция разделяется на большое число немногочисленных подпопуляций, а операторы кроссинговера и селекции применяются лишь в ограниченной области, определяемой отношением соседства. Это приводит к тому, что хорошие решения очень медленно распространяются в популяции (диффузия) и проблема попадания в «локальную яму» становится менее острой.

Ответ на вопрос о том, какой из представленных способов организации распределенных ЭВ является более эффективным, зависит от архитектуры вычислительной системы, используемой в САПР. Решающим обстоятельством для оценки эффективности является то, насколько успешно решается та или иная проектная задача с точки зрения качества решения и вычислительной сложности.

Рекомендации по распараллеливанию операторов базового цикла эволюционных вычислений. Из анализа базового цикла алгоритма ЭВ (см. рис. 1) вытекают следующие рекомендации по распараллеливанию, входящих в него операций.

Рекомендация 1. Две составляющие базового цикла алгоритма ЭВ потенциально могут значительно повлиять на параллелизм: оценка качества решения и репродукция новых решений.

Рекомендация 2. Процесс вычисления целевой функции для каждого решения может осуществляться параллельно и независимо от остальных решений.

Рекомендация 3. Оценка качества решений может потребовать сложных расчетов или моделирования и является одним из наиболее трудоемких этапов ЭВ. В процессе оценки качества требуется сравнить значения функции качества каждого решения с остальными и обновить структуры данных. Это затрудняет процесс распараллеливания.

Рекомендация 4. Выполнение фазы селекции в базовом цикле алгоритма ЭВ может потребовать доступа к определенным подмножествам решений в популяции или обновления данных. Поэтому возможности ее распараллеливания зависят от выбора схемы селекции.

Рекомендация 5. Выполнение фазы репродукции в базовом цикле алгоритма ЭВ нетрудно распараллелить. Репродукция может включать либо копирование решения, либо выполнение унарного оператора (например, мутации отдельной хромосомы), либо выполнение бинарного (например, кроссинговера двух хромосом) или n -арного оператора. Выполнение репродукции является независимой задачей.

На рис. 2 в общих чертах представлены потенциальные возможности распараллеливания базового цикла алгоритма ЭВ.

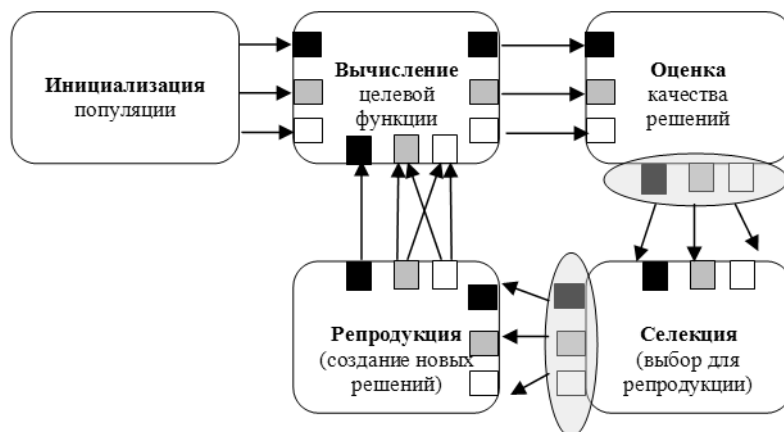


Рис. 2. Потенциальные возможности распараллеливания базового цикла алгоритма ЭВ

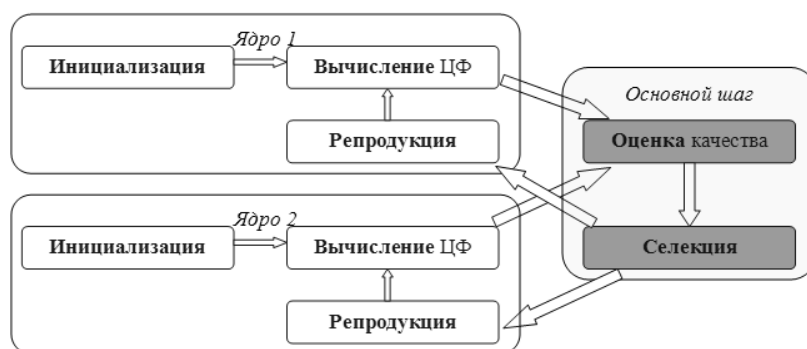


Рис. 3. Параллельное выполнение алгоритма ЭВ на двухядерной машине

В качестве примера на рис. 3 приводится схема параллельного выполнения базового цикла алгоритма ЭВ на двухядерной машине.

Однако поскольку объём популяции решений может быть значительно больше, чем число имеющихся процессоров, то один общий поток заданий будет распределяться по процессорам, что неизбежно приведет к образованию очереди заданий и необходимости организации процессов их обслуживания. Такой подход по эффективности может даже уступать обработке заданий на однопроцессорной машине.

Распределенный алгоритм будет эффективным только в том случае, если задержки, вызванные передачей и обменом необходимыми данными, будут незначительными. Например, если математическая функция \sqrt{x} вычисляется на одном

процессоре, а подкоренное выражение x – на другом, то это займет гораздо больше времени, чем локальное вычисление x и \sqrt{x} .

Уточним возможные варианты организации распределенных вычислений для ЭВ, инспирированных природными системами. Наиболее простым подходом к организации распределенных вычислений представляется глобальный параллелизм и клиент-серверная схема его реализации. Роль клиента выполняет мастер-процессор, реализующий выполнение основного шага, как это показано на рис. 3. Серверы выполняют вычисление целевой функции и репродукцию, которые могут включать несколько ядер.

Однако, как было отмечено выше, для решения проектных задач оптимизации и структурного синтеза большой размерности клиент-серверная архитектура может оказаться не оптимальной. Это произойдет в следующих случаях:

- ◆ в процессе решения задачи время, необходимое для передачи и обмена данными между клиентом и серверами значительно превышает время, необходимое серверу для выполнения своих задач;
- ◆ представление решений и объем популяции требует значительной памяти, поэтому организация распределенного алгоритма ЭВ строится на основе миграционной или диффузионной модели параллелизма.

Используя ЭВ, инспирированных природными системами, можно моделировать эволюцию в «островных подпопуляциях» на различных узлах в сети компьютеров, каждый из которых представляет один «остров» (крупнозернистый параллелизм). Это позволяет преодолеть отмеченные выше недостатки клиент-серверной архитектуры. Связь между узлами понадобится лишь для миграции между «островами». Эта связь может осуществляться асинхронно через сеть самостоятельно работающих алгоритмов ЭВ. Правила миграции могут быть выбраны таким образом, чтобы оптимизировать сетевой трафик. Память, требуемая для организации ЭВ на одной машине, уменьшается, каждый узел работает независимо, как это показано на рис. 4.

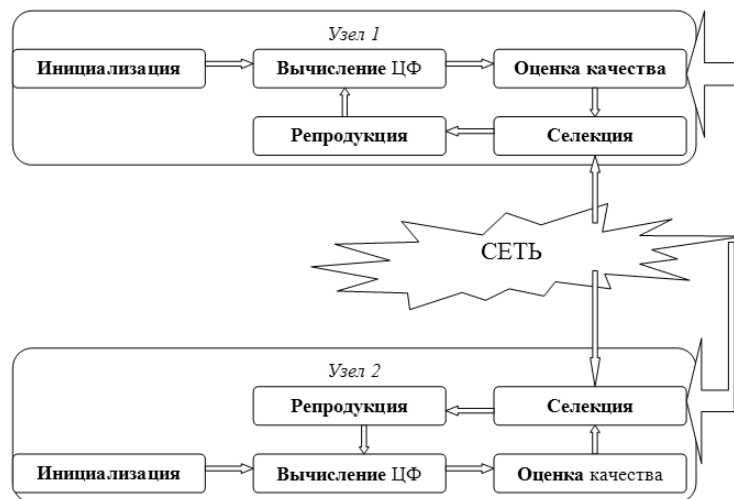


Рис. 4. Параллельное выполнение алгоритмов ЭВ на узлах сети

Возможны и другие и другие способы организации параллельных ЭВ. В частности, в зависимости от задачи периодически можно производить миграцию проектных решений между узлами. Причем в этом случае на различных узлах могут

вычисляться различные целевые функции, что, в свою очередь, открывает возможности для решения в процессе проектирования трудных задач многокритериальной оптимизации.

Заключение. На различных узлах сети могут быть распределены различные алгоритмы эволюционных вычислений, инспирированных природными системами. Гетерогенная смесь этих алгоритмов, имеющих общий базовый цикл выполнения композиции эволюционных операторов, будет более эффективной, нежели гомогенный поиск.

Главные преимущества предлагаемого подхода к организации в интеллектуальных САПР параллельных эволюционных вычислений, инспирированных природными системами, заключаются в простоте его реализации, высокой скорости и робастности поиска проектных решений, в широких возможностях распараллеливания операторов эволюции. При этом скорость поиска будет определяться временем работы эволюционного алгоритма и временем достижения решением заданного качества, а робастность будет означать устойчивость поиска к попаданию в точки локальных экстремумов и способность постоянно от поколения к поколению увеличивать качество популяции решений.

Отметим, что робастность будет во многом зависеть от когнитивных возможностей используемых эволюционных операторов, а наибольшее влияние на скорость эволюционных вычислений, оказывают размеры популяций и хромосом, а также такие особенности целевых функций качества проектных решений как многоэкстремальность (множество ложных аттракторов), «обманчивость», изолированность, дополнительный «шум».

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Курейчик В.М., Родзин С.И. Эволюционные алгоритмы: генетическое программирование (обзор) // Известия РАН. Теория и системы управления. – 2002, № 1. – С. 127-137.
2. Родзин С.И. Интеллектуальные системы. О некоторых алгоритмах, инспирированных природными системами: коллективная монография / под ред. В.М. Курейчика. – М.: Физматлит, 2009. – С. 34-45.
3. Родзин С.И. Мягкие параллельные вычисления // Новости искусственного интеллекта. – 2005, № 4. – С. 53-58.

Родзин Сергей Иванович

Технологический институт федерального государственного образовательного учреждения высшего профессионального образования «Южный федеральный университет» в г. Таганроге.

E-mail: rodzin@mopevm.tsure.ru.

347928, г. Таганрог, пер. Некрасовский, 44.

Тел.: 8(8634)371-673.

Кафедра математического обеспечения и применения ЭВМ; профессор.

Rodzin Sergey Ivanovich

Taganrog Institute of Technology – Federal State-Owned Educational Establishment of Higher Vocational Education “Southern Federal University”.

E-mail: rodzin@mopevm.tsure.ru.

44, Nekrasovskiy, Taganrog, 347928, Russia.

Phone: 8(8634)371-673.

Department of Software Engineering; professor.