

ПАРАЛЛЕЛЬНЫЕ ГЕНЕТИЧЕСКИЕ АЛГОРИТМЫ

Скобцов Ю.А., Эль-Хатиб А.И.

Донецкий национальный технический университет, г.Донецк

кафедра автоматизированных систем управления

E-mail: skobtsov@kita.dgtu.donetsk.ua

Abstract

Skobtsov Y.A., El-Hatib A.I. Parallel genetic algorithms. A survey of modern parallel genetic algorithms is presented in the article. The distributed, cellular and coevolutionary genetic algorithms are considered. The different aspects of parallel genetic algorithms – structurization, parallel implementation, tools for parallelizing are presented.

Простой генетический алгоритм

Генетические алгоритмы (ГА), основанные на формализации принципов естественной эволюции, в настоящее время успешно применяются при решении различных сложных задач и постоянно развиваются [1]. ГА представляют собой алгоритмы случайного направленного поиска для построения (суб)оптимального решения данной проблемы, который моделирует процесс естественной эволюции. Классический “простой” ГА использует двоичные строки для кодирования решения проблемы – особи (хромосомы) [1]. На множестве решений определяется целевая (фитнесс) функция (ЦФ), которая позволяет оценить близость каждой особи к оптимальному решению. Простой ГА использует три основных оператора: репродукция, кроссинговер-скрещивание, мутация. Используя данные операторы, популяция $P(t)$ (множество решений данной проблемы) эволюционирует от поколения к поколению, что можно представить следующей последовательностью действий. Решение задачи на основе простого (классического) ГА можно представить следующей последовательностью действий.

1. Создание исходной популяции.
2. Выбор родителей для процесса размножения (работает оператор отбора - репродукции).
3. Создание потомков выбранных пар родителей (работает оператор скрещивания - кроссинговер).
4. Мутация новых особей (работает оператор мутации).
5. Расширение популяции новыми порожденными особями.
6. Сокращение расширенной популяции до исходного размера (работает оператор редукции).

Если критерий останова алгоритма выполнен, то выбор лучшей особи в конечной популяции – результат работы алгоритма. Иначе переход на шаг 2.

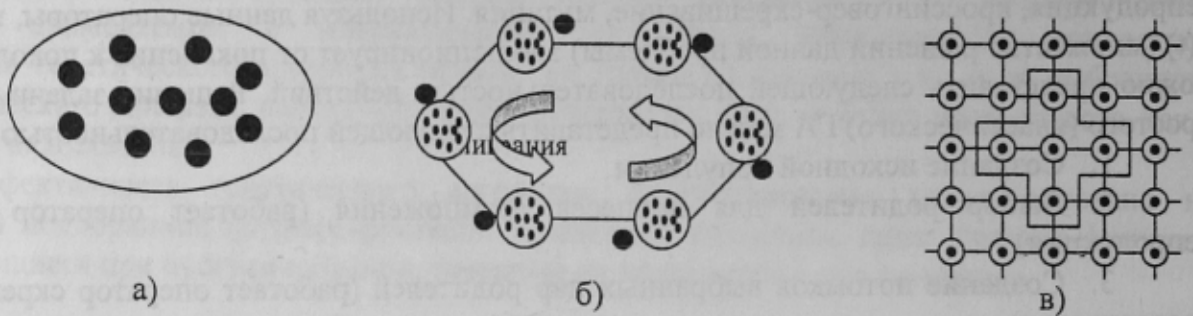
В настоящее время предложены многочисленные модификации и обобщения ГА [2,3], среди которых параллельные ГА (ПГА) развиваются наиболее бурно в теоретическом плане и имеют наиболее важные практические приложения.

Структуризация ГА

Для нетривиальных задач выполнение одного репродуктивного цикла – поколения требует значительных вычислительных ресурсов. При решении многих задач используется не двоичное представление особи – решения проблемы, а более сложные структуры – массивы (матрицы) действительных чисел, связанные списки [2], деревья, графы [3] и т.д. Поэтому вычисление значения фитнес-функции для каждой особи – потенциального решения проблемы, часто является самой трудоемкой операцией в ГА. Для повышения эффективности разрабатываются новые методы кодирования особей, генетические операторы кроссинговера и мутации, гибридные алгоритмы, параллельные алгоритмы и т.п.

Присущий ГА "внутренний" параллелизм и заложенная в них возможность распределенных вычислений способствовали развитию параллельных ГА (ПГА). Первые работы в этом направлении появились в 60-х годах, но только в 80-е годы, когда были разработаны доступные средства параллельной реализации, исследования ПГА приняли систематический массовый характер и практическую направленность. В этом направлении разработано множество моделей и реализаций, некоторые из которых представлены ниже [4-6].

Прежде всего, необходимо отметить, что в основе ПГА лежит структуризация популяции (множества потенциальных решений) – его разбиения на несколько подмножеств (подпопуляций). Это разбиение можно сделать различными способами, которые и определяют различные виды ПГА. Согласно современной классификации различают Глобальные ГА (ГГА) распределенные ГА (РГА), клеточные ГА (КГА) и коэволюционные ГА (КЭГА). В простом ГА, графически представленном на рис. 1.1а), используется одна популяция особей, каждая из которых может взаимодействовать с любой другой особью. На рис.1.1а) каждая особь представлена точкой. В РГА популяция разбивается на множество подпопуляций, каждая из которых эволюционирует независимо (согласно простому ГА) и обменивается через некоторое "время изоляции" с соседними подпопуляциями по определенной схеме, что графически представлено на рис.1.1 б). В КГА имеется множество подпопуляций, каждая из которых состоит только из одной особи. В один момент времени данная особь может взаимодействовать только с соседними особями. Отношение соседства задается в виде некоторой регулярной структуры – сетки, что графически представлено на рис.1.1 г).



- а) простой ГА;
- б) распределенный ГА;
- в) клеточный ГА.

Рисунок 1 - Виды генетических алгоритмов

Распределенные ГА

РГА используют, в основном, так называемую "модель островов", где каждая подпопуляция развивается на своем "острове". Между островами производится (достаточно редко) обмен лучшими особями. Эта модель может быть реализована в распределенной памяти компьютерной системы, имеющей MIMD-архитектуру согласно классификации Flynn. Преимущество РГА в том, что они работают быстрее даже на однопроцессорных компьютерных системах вследствие лучшей структуризации. Причина заключается в том, что число вычислений сокращается благодаря распределению поиска в различных областях пространства решений. Разработаны различные виды РГА, но практически все они являются вариациями базового алгоритма, который представлен следующим псевдокодом.

Распределенный ГА

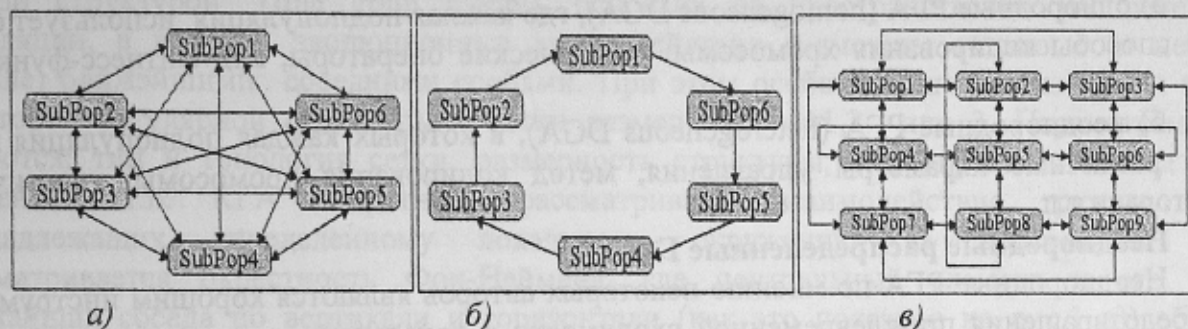
```

{
  Генерация популяции P хромосом случайным образом;
  Разбиение P на подпуляции P1, P2, ..., PN;
  Определение структуры соседства для SPi, I=1, ..., N;
  While(критерий останова не выполнен)
    выполнение для SPi, I=1, ..., N параллельно следующих шагов
    {
      В течение fm поколений выполнение отбора и генетических операторов;
      Посылка nm хромосом в соседние подпопуляции;
      Прием хромосом от соседних подпопуляций;
    }
}

```

Основными факторами, которые влияют на миграцию в модели островов (и следовательно, на их эффективность) являются следующие.

1) **Топология**, определяющая отношение соседства между подпопуляциями. Здесь обмен особями происходит только между соседними подпуляциями. Существует также несколько стандартных схем обмена особями между подпопуляциями, которые представлены ниже.



а) каждая подпопуляция обменивается с каждой из остальных подпопуляций;
 б) обмен по кольцу;
 в) обмен между соседними подпопуляциями на гиперкубе.
 Рисунок 2 – Схемы обмена особями между подпопуляциями

- 2) **Степень миграции**, которая определяет количество мигрирующих особей.
- 2) **Время изоляции**, определяющее число поколений между сеансами миграции.
- 3) **Стратегия отбора особей** в пул обмена. Здесь наиболее распространенными являются два метода. При первом подходе из подпопуляции особи выбираются случайным образом – при этом сохраняется разнообразие генетического материала. При втором методе из каждой подпопуляции выбираются лучшие в некотором смысле особи, что делает процесс более направленным. При этом также существуют различные методы отбора лучших хромосом [2].

4) **Стратегия замены** особей на мигрировавшие хромосомы из соседних подпопуляций. Здесь также существуют различные подходы [2]: из подпопуляции удаляются худшие, случайные особи и т.п.

5) **Стратегия репликации** мигрирующих особей. При первом подходе мигрирующая особь остается также и в "родной" подпопуляции. Второй подход требует удаления мигрирующей особи из "родной" подпопуляции. Впервые стратегия может привести к доминированию в различных подпопуляциях одних и тех же сильных особей.

При второй стратегии особь может через некоторое время вернуться назад в исходную подпопуляцию, что ведет к лишним затратам вычислительных ресурсов.

Виды распределенных ГА

Распределенные ГА можно классифицировать по следующим признакам.

1) По методу миграции принято разделять:

а) изолированные РГА (isolated DGA), где нет миграции между подпопуляциями (иногда называются разделенные РГА – partitioned DGA);

б) синхронные РГА (synchronous DGA), в которых миграции между подпопуляциями синхронизированы – выполняются в одно и тоже время;

в) асинхронные РГА (asynchronous DGA), где миграции могут происходить по событию в разные моменты времени в различных подпопуляциях, в зависимости от активности в каждой подпопуляции (асинхронное поведение характерно для естественной эволюции, которая развивается по разному в зависимости от внешней среды).

2) По изменению схемы обмена особями различают:

а) статическая схема соединений между подпопуляциями, которая не изменяется в процессе эволюции;

б) динамическая схема соединений, в которой вид обмена особями между подпопуляциями может изменяться в процессе эволюции.

б) По однородности подпопуляций принято разделять на:

а) однородные РГА (homogeneous DGA), где каждая подпопуляция использует одни и те же способы кодирования хромосомы, генетические операторы, вид фитнес-функций и т.п.;

б) неоднородные РГА (heterogeneous DGA), в которых каждая подпопуляция может иметь различные параметры управления, метод кодирования хромосомы, генетические операторы и т.п.

Неоднородные распределенные ГА

Неоднородные РГА по мнению некоторых авторов являются хорошим инструментом для предотвращения преждевременной сходимости в локальных экстремумах и позволяют эффективно решать проблему расширения (исследования) и эксплуатации зоны поиска решений. В этой области разработаны некоторые интересные модификации РГА, которые представлены ниже.

1) *Адаптация конкурирующих подпопуляций.* Здесь для каждого возможного вида генетических операторов формируется подпопуляция. Общее число особей фиксировано (во всех подпопуляциях), в то время как мощность каждой подпопуляции может варьироваться. Каждая подпопуляция соревнуется с конкурентами таким образом, что она приобретает или теряет особи в зависимости от ее качества относительно других подпопуляций. Например, разные подпопуляции могут иметь различный шаг мутации (над вещественным представлением) или отличающиеся арифметические операторы рекомбинации. С другой стороны каждая подпопуляция может иметь различные схемы обмена, степень миграции и т.п. Через определенное число поколений происходит ранжирование различных стратегий и параметры каждой адаптируются к значения лучшей подпопуляции.

2) *РГА, основанные на миграции и искусственной селекции (GAMAS).* Эта модификация использует четыре подпопуляции – "породы" I – IV. Сначала создаются "породы" I – IV. Порода II предназначена для исследования (новых) областей поиска (exploration). Для этого в ней применяется мутация с высокой вероятностью $p_m=0,05$. Порода IV используется для эксплуатации найденной зоны поиска решений (exploitation). Поэтому в ней применяется низкая вероятность мутации $p_m=0,003$. Порода III используется как для исследования так и для эксплуатации зон поиска решений и имеет среднее значение вероятности мутации $p_m=0,005$. РГА отбирает лучших особей из пород II-IV и вводит их в

породу I, если они лучше уже имеющихся в этой породе (I). Таким образом, порода I сохраняет лучшие хромосомы, появляющиеся в других породах. Через заданное число поколений ее хромосомы вводятся в породу IV и вытесняют ее текущие особи.

3) *Неоднородные РГА с различным кодированием.*

В этой модификации РГА, иногда называемой "модель островов с инъекцией", в каждой подпопуляции особи кодируются с различной точностью. Это дает возможность исследовать пространство поиска в разных подпопуляциях с различным шагом. Из подпопуляции с "крупной сеткой" лучшие особи "впрыскиваются" в подпопуляцию с "мелкой сеткой". При этом подпопуляция с низкой точностью имеет меньшую размерность пространства поиска и поэтому зона возможного экстремума обычно находится быстрее. Далее лучшие особи "впрыскиваются" в подпопуляцию с большей точностью, где решение уточняется с меньшим шагом.

4) Известны также модификации РГА, в которых в разных подпопуляциях используются арифметические операторы кроссинговера с различной точностью. Здесь, как и в предыдущей модификации, производится впрыскивание лучших решений из "грубой" подпопуляции в "тонкую" для последующей "доводки".

Клеточные ГА

Модель клеточных ГА (cellular GA), часто называемой также диффузией или "модель с тонкой структурой" (fine grain model), основана на пространственно распределенной популяции, в которой эволюционные взаимодействия возможны только с (в некотором смысле) ближайшими соседними особями. При этом особи обычно расположены в узлах некоторой регулярной структуры – сетки размерности $d=1,2$ или 3. Параметрами КГА являются: тип и топология сетки, размерность структуры, тип окрестности, вид отбора особей и т.п. КГА итеративно рассматривают взаимодействие группы особей, принадлежащих определенному локальному окружению. В простейшем случае рассматривается окрестность Фон-Неймана, где центральный элемент и его четыре ближайших соседа по вертикали и горизонтали (как это показано на рис.1 в)) образуют небольшой пул, где применяются генетические операторы. В каждом поколении КГА рассматривает в качестве центрального элемента окрестности только одну особь. Так как особь может принадлежать только нескольким окрестностям то ее изменение влияет на соседей "мягко". Это обеспечивает хороший компромисс между скоростью сходимости и расширением пространства поиска. По синхронизации взаимодействия соседних элементов различают синхронные и асинхронные КГА. В синхронных КГА по синхросигналу (одновременно) вычисляется все новое поколение и записывается во временную (буферную) популяцию, которая затем заменяет старую популяцию согласно следующему алгоритму.

Синхронный клеточный ГА (ГА)

```

{
  для s=1 до МАКС_ЧИСЛО_ПОКОЛЕНИЙ
  {
    для x=1 до ШИРИНА
    {
      для y=1 до ВЫСОТА
      {
        сосед_список=вычисление_соседей(ГА, позиция(x,y));
        родитель1=выбор(сосед_список);
        родитель2=выбор(сосед_список);
        кроссинговер( $P_c$ , родитель1,родитель2, потомок);
        мутация( $P_m$ , потомок);
      }
    }
  }
}

```

```

    фитнес_особи=вычисл_фитнесс(декодирование(потомок));
    внедрение_особи(времен_популяция, позиция(x,y),особь,
        [если фитнес_особи лучше или всегда]);
}
}
}

```

```

текущ_популяция=врем_популяция;
сбор_стат_данных_популяции(текущ_популяция);
}
}

```

Клеточные ГА часто рассматриваются как стохастические клеточные автоматы, где мощность множества состояний равна числу точек в пространстве поиска решений. В синхронном КГА все клетки формально одновременно изменяют свое значение. Разработаны также и асинхронные КГА, где изменение клеток происходит "по событию" – изменению соседних элементов структуры.

Козволюционные ГА

Данный тип ГА заимствует у природы явления кооперации и конкуренции использует обычно две подпопуляции (в общем случае число подпопуляций может быть и больше). Разработаны несколько видов коэволюционных ГА, которые разделяются на кооперативные и конкурирующие ГА. На практике более распространены кооперативные коэволюционные ГА. Здесь взаимодействие между подпопуляциями осуществляется только за счет оценки значений фитнес-функций (нет обмена генетическим материалом между подпопуляциями как в РГА и КГА). При этом значение фитнес-функции особи зависит от ее способности "сотрудничать" с особями других подпопуляций. Этот подход часто позволяет произвести декомпозицию сложной проблемы на несколько менее сложных задач, каждая из которых решается с помощью ГА. Данный тип ПГА наиболее широко применяется при решении задач многокритериальной оптимизации.

Параллельная реализация ГА

На этом этапе при заданной структуризации ГА реализуется собственно параллелизация. При этом ожидаются следующие преимущества:

- 1) поиск альтернативных решений одной и той же проблемы;
- 2) параллельный поиск из различных точек в пространстве решений;
- 3) допускают хорошую реализацию в виде островов или клеточной структуры;
- 4) большая эффективность поиска даже в случае реализации не на параллельных вычислительных структурах;
- 5) хорошая совместимость с другими эволюционными и классическими процедурами поиска;
- 6) существенное повышение быстродействия на многопроцессорных системах.

Рассмотрим современные основные методы реализации ПГА. Наиболее известной является представленная на рис.3.а) **глобальная параллелизация**. Эта модель основана на простом (классическом) ГА с вычислениями, выполняемыми параллельно. Она быстрее, чем классический ГА, выполняемый последовательно, и обычно не требует баланса по загрузке поскольку на разных процессорах чаще всего вычисляются значения фитнес-функций для различных особей (имеющие примерно одинаковую вычислительную сложность). Исключение составляет генетическое программирование, где различные особи могут сильно отличаться по своей сложности (древовидные или граф-структуры). Эту модель часто называют "**рабочий - хозяин**". Многие исследователи используют пул процессоров для

повышения скорости выполнения алгоритма поскольку независимые запуски алгоритма на различных процессорах выполняются существенно быстрее чем на одном процессоре. Отметим, что в этом случае нет никакого взаимодействия между различными прогонами алгоритма. Это чрезвычайно простой метод выполнения одновременной работы (если это возможно) и он может быть очень полезным. Например, он может быть использован для решения одной и той же задачи с различными начальными условиями. В силу своей вероятностной природы ГА позволяют эффективно использовать этот метод. При этом ничего нового в сам алгоритм по сути ничего не вносится, но выигрыш во времени может быть значительным.

На рис.3 б) представлена также чрезвычайно популярная "модель островов"(coarse grain), где множество подалгоритмов совместно работают параллельно, обмениваясь в процессе поиска некоторыми особями. Эта модель допускает прямую реализацию на компьютерных системах с MIMD- архитектурой. При этом каждый "остров" соответствует своему процессору. Очевидно модель характеризуется параметрами, которые были представлены в разделе 3.

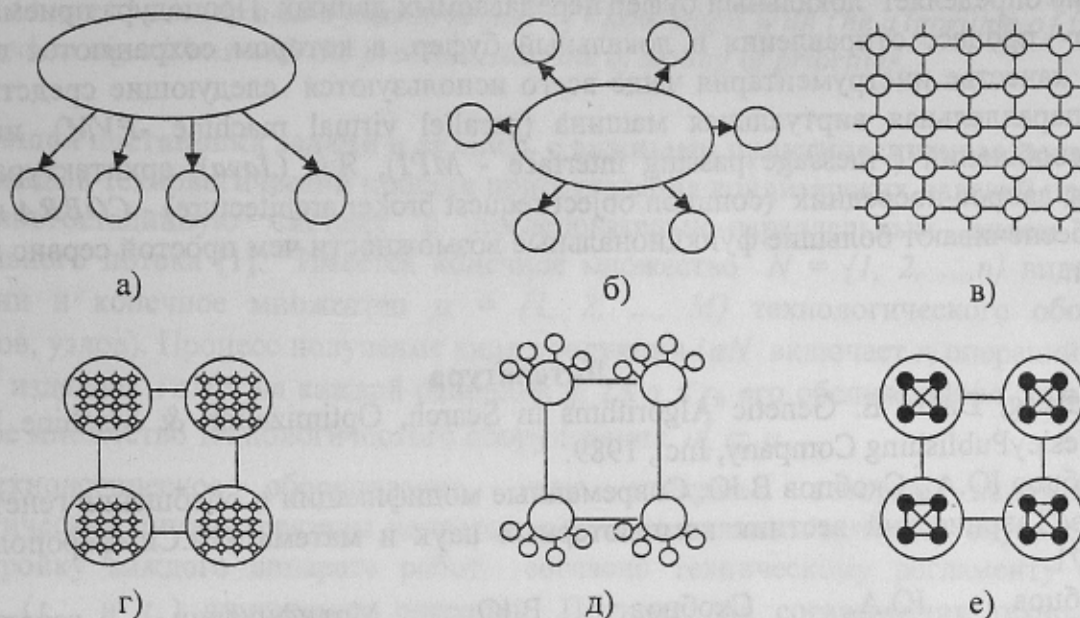


Рисунок 3 - Различная реализация параллельных ГА

В *клеточных ГА (fine grain)*, показанных на рис.3. в), параллелизация обычно реализуется на компьютерных системах с SIMD-архитектурой, где каждый процессор представляет подпопуляцию (из одной особи). Хотя известны работы, в которых используются однопроцессорные компьютеры и системы с MIMD-архитектурой.

На рис.3 г-е) представлены гибридные модели, в которых при параллелизации используется двухуровневый подход. В данном случае на верхнем уровне для параллелизации применяется РГА (coarse-grain implementation). В модели рис.3.г) каждый из островов реализуется (на нижнем уровне) в виде клеточного ГА, что позволяет объединить преимущества этих двух различных подходов. Модель рис.3.д) каждый из островов на нижнем уровне реализуется по схеме глобальной параллелизации - "рабочий - хозяин". Таким образом, здесь параллелизм используется для ускорения вычислений (на нижнем уровне) и в тоже время он применяется при реализации взаимодействующих подпопуляций - островов (на верхнем уровне). И, наконец, в модели рис.3.е) на верхнем и нижнем уровнях реализуется модель островов. Таким образом, как классический ГА, так и

структурированный ГА могут быть реализованы различными способами как на монопроцессорной так и на многопроцессорных компьютерных системах.

Инструментарий параллелизации

Как отмечалось выше, для реализации ПГА могут быть использованы компьютерные системы с различными архитектурами: SISD, SIMD, MIMD и т.д. Вместо описания многочисленных специальных архитектур и программных конструкций, которые используются при реализации ПГА, далее мы кратко рассмотрим те параллельные и распределенные модели, которые не зависят от конкретных структур. Почти все ПГА реализуются на основе модели передачи сообщений коммуникации, поскольку она позволяет описывать многопроцессорные системы с распределенной памятью, которые являются наиболее удобным и распространенным средством реализации ПГА. В модели передачи сообщений процессы в одном или физически различных процессорах общаются между собой путем передачи друг другу сообщений через среду коммуникации, которая представлена стандартом или специальной схемой соединения. Основными элементами здесь являются процедуры отправления и приема сообщений. В простейшей форме, отправление определяет локальный буфер передаваемых данных. Процедура приема обычно определяет процесс отправления и локальный буфер, в котором сохраняются входящие данные. В качестве инструментария чаще всего используются следующие средства: сокет (*sockets*), параллельная виртуальная машина (*parallel virtual machine -PVM*), интерфейс передачи сообщений (*message passing interface - MPI*), Ява (*Java*), архитектура общего назначения запрос-посредник (*common object request broker architecture - COBRA* и *Globus*, которые обеспечивают большие функциональные возможности чем простой сервис передачи сообщений.

Литература

5. Goldberg, David E. Genetic Algorithms in Search, Optimization & Machine Learning. Addison-Wesley Publishing Company, Inc., 1989.
6. Скобцов Ю.А., Скобцов В.Ю. Современные модификации и обобщения генетических алгоритмов//Таврический вестник компьютерных наук и математики. Симферополь: 2004, N1. – с.60-71.
7. Скобцов Ю.А., Скобцов В.Ю. Модификации генетического программирования//Труды конференций "Интеллектуальные системы" и "Интеллектуальные САПР". -Москва: Физматлит. - 2004. - с.76-81.
8. P. Adamadis. Review of parallel genetic algorithms bibliography. Tech. Rep., Aristotle Univ., Thessalonik, Greece. http://www.control.ee.auth.gr/panos/papers/pgs_review.ps.gz.
9. E. Alba, J.M. Troya.. A survey of parallel distributed genetic algorithms//Complexity, vol.4, no.4, pp.31-52, 1999.
10. E. Alba, M. Tomassini. Parallelism and evolutionary algorithms//IEEE trans. On evolutionary computation. -2002. -vol.6. N5. -pp.443-462.