

УДК 004.8+004.42

## ПРАКТИЧЕСКИЕ АСПЕКТЫ ИСПОЛЬЗОВАНИЯ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ В ГЕНЕТИЧЕСКОМ АЛГОРИТМЕ

В.В. ФРОЛОВ

*Харьковский национальный университет имени В.Н. Каразина, Украина*

*Поступила в редакцию 7 февраля 2018*

**Аннотация.** Рассмотрены практические приемы реализации параллельных вычислений в генетическом алгоритме для дискретной оптимизации на примере расчета размерных цепей. Предложен подход к определению граничных размеров популяции на основе расчета вероятности независимых случайных событий.

*Ключевые слова:* генетический алгоритм, популяция, кроссовер, мутация.

**Abstract.** Practical implementation methods for parallel computations in the genetic algorithm for discrete optimization, using the example of calculating dimensional chains are considered. An approach for determining the boundary dimensions of population on the basis of probability calculating for independent random events is proposed.

*Keywords:* genetic algorithm, population, crossover, mutation.

**Doklady BGUIR. 2018, Vol. 115, No. 5, pp. 24-30**

**Practical aspects of using of parallel computing in genetic algorithm**

**V.V. Frolov**

### Введение

Одним из основных недостатков генетических алгоритмов (ГА) является его низкая производительность по сравнению со специальными методами оптимизации, которые учитывают особенности решения конкретной оптимизационной задачи. На основе анализа подходов к повышению производительности ГА можно выделить несколько основных направлений:

1. Использование параллельных вычислений [1–3].
2. Комбинирование генетических алгоритмов с другими методами оптимизации, например, выполнение глобального поиска с помощью ГА, а затем использование специализированных методов оптимизации [1, 4].
3. Модификация основных операторов ГА [1].
4. Модификация всего ГА для повышения эффективности решения конкретной задачи [2, 5].
5. Совместное использование ГА и искусственных нейронных сетей [4].
6. Выбор оптимальных параметров ГА, таких как размер популяции, количество эволюций (эпох), вероятность мутации и кроссовера на основе анализа особенностей решения конкретной задачи [5, 6].

Таким образом, если совместить несколько из указанных направлений, то эффект повышения производительности должен получиться кумулятивным. В связи с этим целью данной работы является оценка возможности повышения производительности ГА заданной структуры для решения задач дискретной оптимизации за счет использования параллельных вычислений и предварительного выбора наиболее рационального размера популяции с обоснованием

необходимого количества эволюций. Такой подход, с одной стороны, должен ускорить вычисления за счет более полного использования мощностей многоядерных процессоров, а с другой – за счет определения наиболее рациональной выборки из области поиска позволить найти решение быстрее с заданным уровнем надежности.

Для достижения цели необходимо решить следующие задачи: выбрать способ реализации параллельных вычислений; разработать метод определения размера популяции для дискретных задач.

### 1. Анализ структуры существующего генетического алгоритма

В работе [7] предлагается ГА для решения дискретных задач в машиностроении. В частности, будем рассматривать применение ГА для решения задач расчета линейных размерных цепей [8], которые представлены с помощью нейронных сетей прямого распространения без обратных связей.

Основные структурные элементы ГА, согласно [7], можно представить следующим набором методов в порядке их следования:

1. Формирование популяции – «Популяция».
2. Событие о проценте выполнения алгоритма – «Прогресс».
3. Вычисление средней функции популяции – «Средняя».
4. Селекция родителей – «Родители».
5. Скрещивание родительских хромосом – «Кроссовер».
6. Мутация родительских хромосом – «Мутация».
7. Вычисления функции приспособленности потомков – «Функция».
8. Формирование новой популяции – «Репродукция».

Селекция родителей происходит следующим образом: лучший со всеми; самый близкий к случайному; самый близкий к лучшему; случайный; самый дальний от случайного; лучший с худшим. Это позволяет получить более разнообразный материал для формирования родительских пар.

В начале выполнения каждого из указанных методов стартуем секундомер и останавливаем его после окончания работы, таким образом, получаем время в миллисекундах, необходимое для определения возможности использования приемов асинхронного программирования. Для контроля правильности замера времени замеряем общее время выполнения каждой итерации (эволюции), назовем его – «Эпоха» ( $T_{epoch}$ ). Если выполняется условие (1), т. е. общее время не меньше суммы всех времен элементов, составляющих ГА, то замер выполнен правильно.

$$T_{\Sigma} = \sum_{i=1}^8 T_i, T_{\Sigma} \leq T_{epoch}. \quad (1)$$

Проводим численный эксперимент для замеров времени последовательного выполнения ГА со следующими параметрами: количество эволюций – 1000; размер популяции – 200; вероятность кроссовера – 0,95; вероятность мутации – 0,05; вероятность смены популяции при стабилизации функции приспособленности – 0,98; проверка стабилизации начинается с 250 эволюции; глубина проверки на стабилизацию – 50 эволюций.

Функция приспособленности на рис. 1 представлена нейронной сетью прямого распространения без обратных связей с одним скрытым слоем, состоящим из двух нейронов со степенными функциями активации, и одним выходным нейроном с линейной функцией активации. Ее можно записать аналитически в виде

$$y = K \times \left[ \left[ \left[ \left( \sum_{i=1}^7 w_{1,i} \cdot x_i \right) + b_1 \right]^2 + \left[ \left( \sum_{i=1}^5 w_{2,i} \cdot x_{i+6} \right) + b_2 \right]^2 \right] + b_3 \right], K = 1. \quad (2)$$

Значения входов заданы множествами значений следующим образом:

$$x_i = \{a_1, a_j, \dots, a_{12}\}, a_j \in \mathbb{Z}, j = \overline{1,12}.$$

Следовательно, область поиска будет равна прямому произведению этих множеств:

$$X = \bigotimes_{i=1}^{11} x_i, |X| = 12^{11}.$$

По мощности результирующего множества  $|X|$  области поиска и форме функции  $u$  задача имеет достаточную сложность для применения ГА.

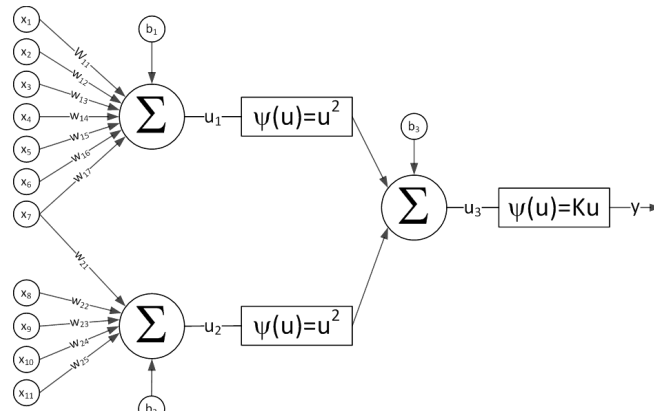


Рис. 1. Схема функции приспособленности в виде нейронной сети:

$x$  – вход;  $w$  – синапс;  $b$  – смещение;  $u$  – результат сумматора;  $\psi$  – функция активации;  $y$  – выход сети

## 2. Анализ производительности существующего генетического алгоритма

В результате имеем по 1000 замеров времени на каждый элемент из условия (1). Рассчитываем среднее по всем замерам и заносим в табл. 1 со статистическими параметрами: доля во времени эпохи в процентах;  $R$  – размах;  $S$  – дисперсия;  $\sigma$  – среднеквадратическое отклонение.

Таблица 1. Статистика по замерам времени выполнения методов ГА

Название метода	Имя	Время, мс	Доля, %	max, мс	min, мс	$R$	$S$	$\sigma$
Популяция	$T_1$	0,2726	<b>0,7718</b>	272,6401	0	272,6401	74,3326	8,6216
Прогресс	$T_2$	0,0003	0,0007	0,0207	0	0,0207	$1 \times 10^{-6}$	0,0009
Средняя	$T_3$	0,0130	0,0367	0,0572	0,0119	0,0453	$5 \times 10^{-6}$	0,0022
Родители	$T_4$	32,0897	<b>90,8436</b>	100,9926	0,0007	100,9919	1756	41,9061
Кроссовер	$T_5$	0,0877	0,2482	1,6389	0	1,6389	0,0173	0,1314
Мутация	$T_6$	0,0005	0,0014	0,0549	0	0,0549	$5 \times 10^{-6}$	0,0022
Функция	$T_7$	2,7616	<b>7,8178</b>	4,8733	0	4,8733	0,0345	0,1857
Репродукция	$T_8$	0,0856	0,2423	0,3598	0,0733	0,2865	0,0002	0,0152
Итого		35,311	99,9625					
Эпоха	$T_{epoch}$	35,324		272,7768	2,5259	270,2509	1812	42,5763

Анализ показывает самые длительные методы, для которых нужно использовать параллельные вычисления, чтобы ускорить выполнение ГА: популяция, родители, функция. Суммарная доля этих методов в одном такте ГА составляет 99,43 %, следовательно, по закону Адмала в формулировке работы [9] получаем значение ускорения вычислений за счет параллельных вычислений по формуле

$$Speedup \leq \frac{1}{(1 - pctPar) + \frac{pctPar}{p}}, \quad (3)$$

где  $Speedup$  – ускорение за счет распараллеливания;  $pctPar$  – доля времени, где будут выполняться параллельные вычисления, в нашем случае 0,9943;  $p$  – количество ядер процессора, в нашем случае 2 ядра.

Вычисления по формуле (3) дают следующий результат:

$$Speedup \leq \frac{1}{(1 - 0,9943) + \frac{0,9943}{2}} \Rightarrow Speedup \leq 1,989$$

Самым длительным методом является метод, который выполняет селекцию родителей. Здесь используется множество методов выбора для получения разнообразия скрещиваемого материала, поэтому необходимо определить, где конкретно затраты времени максимальны, поскольку каждый элемент в этом методе выполняется с равной вероятностью. Для этого проведем кластерный анализ методом  $k$ -средних для результатов замеров эпох и сравним со статистикой выполнения операторов селекции во всем ГА. Результаты анализа приведены на рис. 2. Здесь все времена выполнения тактов ГА разбиваются на три кластера в диапазонах: от 0 до 4мс; от 50 до 100мс; от 200 до 300мс.

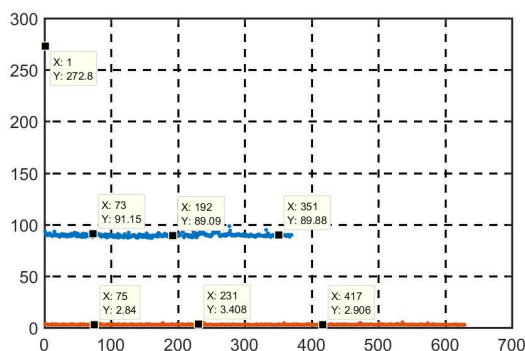


Рис. 2. Разбивка по кластерам времени выполнения каждого такта ГА

Сведем статистику выполнения операторов селекции и кластерного анализа в табл. 2.

Таблица 2. Сравнительный анализ для операторов селекции

Наименование	Центр кластера	Количество	Итого
Кластерный анализ			
Кластер 1	89,7081	370	370
Кластер 2	2,956	629	
Кластер 3	272	1	630
Итого		1000	1000
Выбор родителей			
Лучший со всеми		76	630
Самый близкий к случайному		61	370
Самый близкий к лучшему		62	
Случайный		266	
Самый дальний от случайного		247	
Лучший с худшим		288	
Итого		1000	1000

В результате имеем, что в первом кластере, где центром является длительность выполнения около 89 мс, собрано 370 выполнений. Этому количеству соответствует совокупность следующих операторов: самый близкий к случайному; самый близкий к лучшему; самый дальний от случайного. Эти операторы используют Евклидово расстояние как меру близости. Вычисление данной метрики для всей популяции и определяет основные затраты времени. Алгоритм вычисления метрики позволяет разбить вычисления по нескольким ядрам. Таким образом, основные позиции алгоритма, где можно применить параллельные вычисления, выделенные в табл. 1: популяция, родители, функция.

Особенностью данного алгоритма является то, что при «зависании» в локальных минимумах, которое характеризуется стабилизацией средней функции приспособленности в популяции, происходит смена популяции. При исследовании времени выполнения смена популяции производилась один раз. Но может возникнуть ситуация, когда смена будет производиться много раз для обеспечения выхода из локального минимума, тогда затраты времени на формирование популяции, судя по соотношению времен центра кластеров в табл. 2, могут иметь значительную долю вычислений в рамках одной эпохи, поэтому этот элемент также включаем в стратегию распараллеливания.

### 3. Организация параллельных вычислений

Данный ГА реализован в программе расчета размерных цепей на платформе .NET. Следовательно, будем использовать методы классов Task и Parallel платформы для организации вычислений в нескольких потоках. Схема организации вычислений выполнена по способу «Master-Slave» [3]. Здесь в качестве мастера выступает основной цикл ГА, а подчиненными будут: метод вычисления функции приспособленности для каждого потомка, поскольку потомков два, организуем два потока для вычисления каждого; метод формирования популяции, здесь популяция разбивается по потокам в зависимости от количества ядер процессора; метод вычисления Евклидовой метрики для особей популяции, аналогично разбивается по потокам в зависимости от количества ядер процессора.

При этом учитывается теорема о зависимости по данным [10] для каждого потока, чтобы обеспечить максимальную производительность. Тогда общий подход к формированию параллельно выполняемых потоков следующий.

1. Выполняем локализацию данных для каждого потока, чтобы избежать блокировки общих данных отдельными потоками.
2. Формируем действия в потоках.
3. Запускаем потоки «Slave».
4. По завершении всех потоков собираем локализованные данные и передаем их «Master» (основному ГА).

В результате имеем среднюю длительность выполнения эпохи 21,6585829 мс. Из отношения последовательного выполнения к параллельному, с учетом двух ядер, получаем ускорение  $35,3241/21,6586 = 1,63$ . Следовательно, решив обратную задачу для формулы Адмала, получим долю последовательного кода, которую удалось распараллелить, – 0,77. Если отслеживать на рис. 3 нагрузку ядер процессора при последовательном и параллельном выполнении ГА, то видим что ускорение реализуется за счет более полного использования возможностей ядер: 49 % против 92 %, что доказывает эффективность процедур распараллеливания.

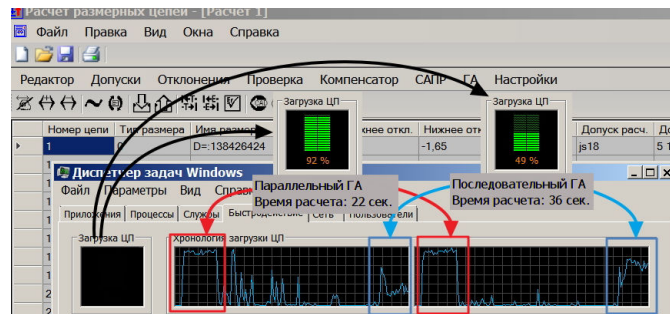


Рис. 3. Нагрузка ядер процессора при параллельном и последовательном вычислениях

### 4. Определение размера популяции для дискретной оптимизации

Считается, что для эффективного поиска решений в сформированной популяции должны быть гены со всеми допустимыми значениями. Тогда будет необходимый материал для скрещивания, который позволит охватить нужную область поиска, содержащую оптимальное решение. Для решения задач дискретной оптимизации имеем конечную генеральную совокупность сочетаний значений, в которой находится оптимальное решение. В примере, выбранном для данной статьи, эта область поиска равна мощности множества  $|X|$ . Тогда определить размер популяции возможно с учетом вероятности появления оптимального значения из формулы Бернулли, поскольку появление значения в каждой переменной является случайным, не зависящим от других переменных. Определим функцию расчета вероятности (4) положительных  $k$  исходов при  $n$  независимых испытаниях по формуле Бернулли

$$P(n, k, p) = \frac{\prod_{i=1}^k [(n-k) + i]}{k!} \cdot p^k \cdot (1-p)^{n-k}, \quad (4)$$

где  $p$  – вероятность появления одного числа из заданного множества. В нашем случае эта вероятность равняется  $Pm = 1/12$ .

Если задаться границами изменения исходов и испытаний, можно построить график изменения функции (4) (рис. 4, а).

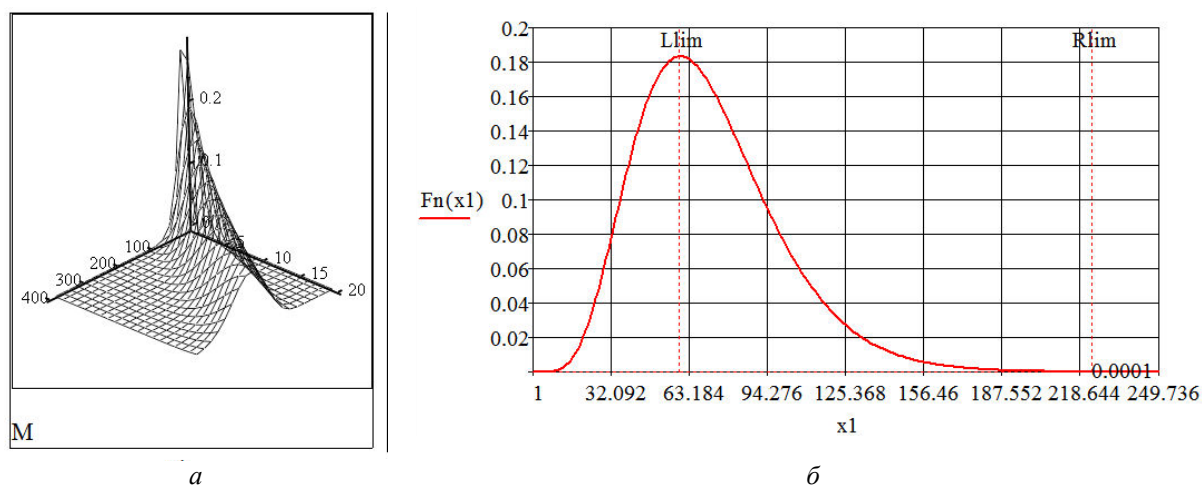


Рис. 4. Функция вероятности положительных исходов:  
а – общий вид функции; б – сечение функции при  $k = 5$ .

На рис. 4, а виден экстремальный характер функции (4) в определенном сечении при заданном количестве положительных исходов. Следовательно, можно определить границы изменения объема популяции, которые учитывают возможность использования всех значений из множеств. При этом объем популяции будет равен количеству испытаний  $n$  при заданном количестве положительных исходов. Например, для  $k=5$  сечение функции (4) будет представлено на рис. 4, б. По графику на рис. 4, б можно определить минимально необходимую границу  $Llim = 59,499$  и предельную границу объема популяции  $Rlim = 223,319$  особи.

### Заключение

Эффективность распараллеливания однопопуляционного ГА определяется особенностями его структуры и реализацией основных генетических операторов на основе анализа времени выполнения каждого оператора. Потенциально при решении задач выбора необходимого размера популяции возможно получить более значительное ускорение работы алгоритма за счет раннего нахождения оптимального решения, что определяется качественным составом особей в популяции. Другими словами, чем более разнообразным будет материал для скрещивания, тем выше вероятность быстрого нахождения оптимума.

### Список литературы

1. Емельянов В.В., Курейчик В.В., Курейчик В.М. Теория и практика эволюционного моделирования. М.: ФИЗМАТЛИТ, 2003. 432 с.
2. Кононюк А.Е. Дискретно-непрерывная математика. (Алгоритмы): в 12 кн. Кн. 10 Алгоритмы. Ч. 3: Генетические алгоритмы. Киев: Освіта України, 2017. 444 с.
3. Курейчик В.М., Кныш Д.С. Параллельный генетический алгоритм. Модели и проблемы построения. [Электронный ресурс]. URL: <http://www.raai.org/resurs/papers/kolomna2009/doklad/Kureichik.doc> (дата обращения: 27.12.2017)
4. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. М.: Горячая линия Телеком, 2006. 452 с.
5. Семенычев Е.В., Куркин Е.И., Данилова А.А. Выбор параметров генетических алгоритмов в задачах параметрической идентификации нелинейных моделей динамики // Вестн. Самарского муниципального института управления. 2013. № 1 (24). С. 130–140.
6. Тимофеева Н.Е., Савин А.Н. Определение оптимальных параметров генетического алгоритма на основе данных регрессионного анализа численного планируемого эксперимента // Материалы Междунар. науч. конф. «Компьютерные науки и информационные технологии». Саратов, 2016. С. 417–421.

7. Фролов В.В. Особенности реализации генетического алгоритма для проектирования технологических систем механической обработки // Восточно-Европейский журнал передовых технологий. 2012. № 3/9(57). С. 60–64.
8. Фролов В.В. Анализ и расчет размерных цепей деталей автотракторной техники на основе искусственных нейронных сетей // Автомобильный транспорт. Сборник научных трудов. 2010. № 27. С. 104–109.
9. Оценка эффективности параллелизации. Опубликовано от 30 ноября 2011 г. [Электронный ресурс]. URL: <https://software.intel.com/ru-ru/articles/predicting-and-measuring-parallel-performance> (дата обращения: 27.12.2017).
10. Карпов В.Е. Введение в распараллеливание алгоритмов и программ // Компьютерные исследования и моделирование. 2010. Т. 2, № 3. С. 231–272.

### References

1. Emel'janov V.V., Kurejchik V.V., Kurejchik V.M. Teorija i praktika jevoljucionnogo modelirovanija. M.: FIZMATLIT, 2003. 432 s. (in Russ.)
2. Kononjuk A.E. Diskretno-nepreryvnaja matematika. (Algoritmy): v 12-i kn. Kn.10 Algoritmy. Ch. 3: Geneticheskie algoritmy. Kiev: Osvita Ukraïni, 2017. 444 s. (in Russ.)
3. Kurejchik V.M., Knysh D.S. Parallelnyj geneticheskij algoritm. Modeli i problemy postroenija. [Electronic resource]. URL: <http://www.raai.org/resurs/papers/kolomna2009/doklad/Kureichik.doc> (Date of access: 27.12.2017). (in Russ.)
4. Rutkovskaja D., Pilin'skij M., Rutkovskij L. Nejrornyie seti, geneticheskie algoritmy i nechetkie sistemy. M.: Gorjachaja linija Telekom, 2006. 452 s. (in Russ.)
5. Semjonychev E.V., Kurkin E.I., Danilova A.A. Vybor parametrov geneticheskikh algoritmov v zadachah parametricheskoy identifikacii nelinejnyh modelej dinamiki // Vestn. Samarskogo municipal'nogo instituta upravlenija. 2013. № 1 (24). S. 130–140. (in Russ.)
6. Timofeeva N.E., Savin A.N. Opredelenie optimal'nyh parametrov geneticheskogo algoritma na osnove dannyh regressionnogo analiza chislennogo planiruемого jeksperimenta // Materialy Mezhdunar. nauch. konf. «Komp'juternye nauki i informacionnye tehnologii». Saratov, 2016. S. 417–421. (in Russ.)
7. Frolov V.V. Osobennosti realizacii geneticheskogo algoritma dlja proektirovanija tehnologicheskikh sistem mehanicheskoy obrabotki // Vostochno-Evropejskij zhurnal peredovyh tehnologij. 2012. № 3/9(57). S. 60–64. (in Russ.)
8. Frolov V.V. Analiz i raschet razmernih cepej detalej avtotraktornoj tehniki na osnove iskusstvennyh nejrornyh setej // Avtomobil'nij transport. Sbornik nauchnyh trudov. 2010. № 27. S. 104–109. (in Russ.)
9. Ocenka jeffektivnosti parallelizacii. Opublikovano ot 30 nojabrja 2011 g. [Electronic resource]. URL: <https://software.intel.com/ru-ru/articles/predicting-and-measuring-parallel-performance> (Date of access: 27.12.2017). (in Russ.)
10. Karpov V.E. Vvedenie v rasparallelivanie algoritmov i programm // Komp'juternye issledovanija I modelirovanie. 2010. Т. 2, № 3. S. 231–272. (in Russ.)

### Сведения об авторах

Фролов В.В., д.т.н., доцент, профессор кафедры теоретической и прикладной информатики Харьковского национального университета им. В.Н. Каразина.

### Information about the authors

Frolov V.V., D.Sci, associate professor, professor of the department of theoretical and applied informatics of the Kharkov national university named after V.N. Karazin.

### Адрес для корреспонденции

61022, Украина,  
г. Харьков, пл. Свободы, 4,  
Харьковский национальный университет  
им. В.Н. Каразина  
тел. +380675260692;  
e-mail: [vvicfrol@rambler.ru](mailto:vvicfrol@rambler.ru)  
Фролов Вячеслав Викторович

### Address for correspondence

61022, Ukraine,  
Kharkov, Svobody sq., 4,  
Kharkiv national university  
named after V.N. Karazin  
tel. +380675260692;  
e-mail: [vvicfrol@rambler.ru](mailto:vvicfrol@rambler.ru)  
Frolov Vyacheslav Viktorovich