

Адаптивный бионический алгоритм решения задачи о потоке данных минимальной стоимости

Ю.О. Чернышев¹, А.Ю. Полуян², Н.Н. Венцов², П.А. Панасенко¹

¹ Краснодарское высшее военное училище имени генерала армии С.М.Штеменко

² Донской государственный технический университет, Ростов-на-Дону

Аннотация: Представлен адаптивный алгоритм решения задачи о потоке данных минимальной стоимости в статической и динамической постановке. В статической модели задачи параметры, описывающие смежность вершин графа и стоимость передачи данных, являются постоянными, а в динамической – изменяемыми. В случае статической задачи цель алгоритма – получение решения близкого к требуемому, а в случае динамической – адаптация процесса передачи данных под изменяющуюся модель сети. Отличительной особенностью алгоритма является комбинирование методов эволюционной и альтернативной адаптации. Эволюционные методы, такие как микро-, макро-, мета-эволюция обеспечивают локальную корректировку вычислительного процесса, а методы альтернативной адаптации определяют стратегию поиска. Приведен пример задания рекомендуемого потока данных в нечеткой форме, а также способ адаптации процесса передачи данных к рекомендуемым параметрам.

Ключевые слова: поток данных, адаптация, эволюция, оптимизация, эволюционный поиск.

Введение

При построении *статической* модели задачи о потоке минимальной стоимости постоянными являются данные о смежности вершин графа и стоимости передачи сообщений. Две вершины графа i и j соединяются дугой (i, j) только в том случае если из вершины A_i можно передать данные в вершину A_j . Каждой такой дуге назначается вес, соответствующий времени передачи единицы информации от A_i к A_j . Таким образом, матрица смежности $\|a_{i,j}\|_{n,m}$ описывает возможные направления передачи данных, матрица $\|c_{i,j}\|_{n,m}$ предельно допустимое количество данных передаваемых от вершины A_i к A_j . Матрица $\|x_{i,j}\|_{n,m}$ отражает количество данных фактически передаваемых в единицу времени.

Граф, описываемый матрицей $\|a_{i,j}\|_{n,m}$ является ориентированным и может быть циклическим. Так как при решении задач оптимизации удобнее использовать граф, не содержащий циклов, имеющийся циклический граф

можно привести к эквивалентной ациклической форме. Графы подразумеваются эквивалентными в контексте однозначности описываемых ими маршрутов передачи данных. Задачу получения ациклического графа в терминах целочисленного линейного программирования можно трактовать как задачу о покрытии. Из полученного ациклического графа можно получить последовательный, т.е. удовлетворяющий трем свойствам [1,2]:

- граф обладает порядковой функцией $P(x)$ со значениями $0, 1, \dots, n$, где n - количество слоев, на которые разбит граф;
- вершины, находящиеся в одном слое не инцидентны друг другу;
- количество дуг, исходящих из слоя $S_i, i=1, \dots, n-1$ совпадает с количеством дуг, входящих в слой $S_{i+1}, i=1, \dots, n$, причем в слой S_i могут входить лишь дуги из слоев S_k с номерами $k < i, k = 1, 2, \dots, i-1$.

Модернизируем матрицу $\|a_{i,j}\|_{n,m}$, добавив в качестве $(m+1)$ -го столбца вектор столбец, координаты которого вычисляются на основе формулы:

$$a_{i,m+1} = \sum_{j=1}^n a_{i,j},$$

где $i=1, 2, \dots, n$.

Вершины k для которых $a_{k,m+1} = 0$ не имеют исходящих дуг и образуют первый слой S_1 для которого справедливо $P(S_1)=1$.

Элементы вектора столбца $a_{k,m+2}$ определяются по формуле:

$$a_{i,m+2} = a_{i,m+1} - \sum_{a_j \in S_1} a_{i,j},$$

где $i=1, 2, \dots, n$.

Второй слой S_2 составляют нулевые значения столбца $a_{k,m+2}$. Для вершин, образующих слой S_2 , справедливо $P(S_2)=2$. Аналогично вычисляются последующие слои, общее количество всех слоев не превышает n .

На основе матрицы $\|a_{i,j}\|_{n,m}$ и функции P вводятся фиктивные вершины и дуги. Если $a_{i,j} = 1$, т.е. от вершины a_i дуга ведет к вершине a_j и $P(a_i) < P(a_j)$, то во все промежуточные слои вводятся вершины соединяемые дугами нулевого веса. Первая фиктивная дуга, идущая от a_i к первой фиктивной вершине, обретает вес $c_{i,j}$. Фиктивные дуги и вершины не вводятся в случае, если вершинам i и j соответствуют значения порядковой функции P , отличающиеся на единицу.

В полученном последовательном графе, двудольный граф разбивается на два подграфа. К полученным подграфам применяется метод Форда, исходя из которого, образуются подпопуляции A_1, \dots, A_{Count} формирующие начальные решения. [2,3].

Предмет исследования

На рис. 1 приведена структурная схема алгоритма решения задачи о потоке данных минимальной стоимости. В данной схеме используется концепция интеграции различных моделей эволюции (совмещение микро-, макро-, метаэволюции), реализовано активное взаимодействие с внешней средой и адаптация к внешней среде, поддерживается иерархическое управление бионическим поиском, применение модифицированных генетических операторов, построенных на основе различных эволюционных стратегий и поисковых методов. На рис.1 БП1, БП_Count - унифицированные процедуры бионического поиска для подпопуляций, которые выполняются параллельно и независимо. Периодически происходит обмен особями между подпопуляциями при помощи модифицированного оператора мутации с равновероятностным распределением потомков. Процедуры распараллеливания и обмена могут осуществляться многократно.

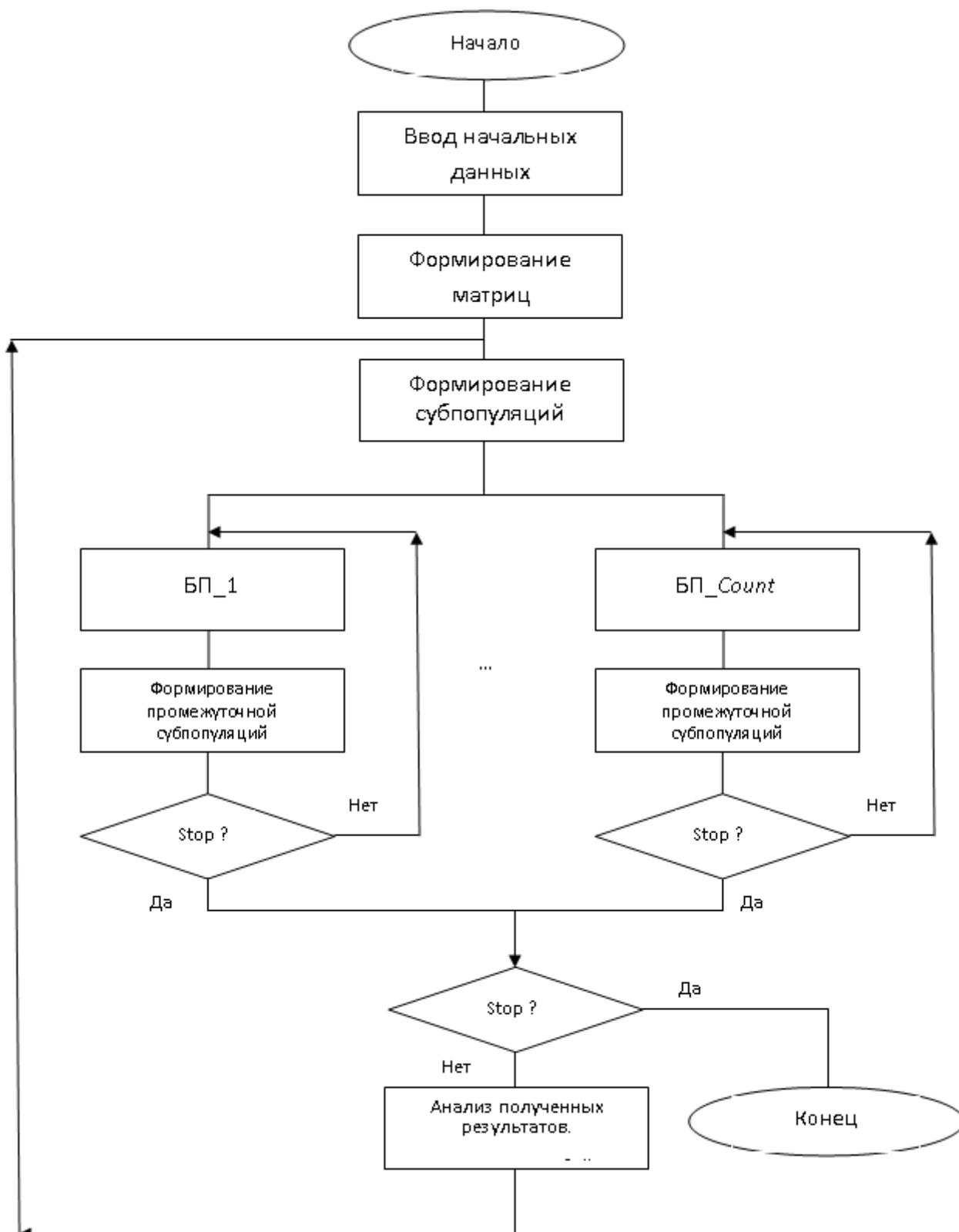


Рис. 1. – Структурная схема алгоритма решения задачи о потоке данных минимальной стоимости

Рассмотрим основные этапы бионического поиска [1]:

1. Определение граничных условий, задание критерия оптимальности, вычисление значений целевых функций для особей подпопуляции.
 2. Проверка особей подпопуляции на соответствие условию попадания в «локальные ямы», если условие выполняется, то переход к п.6, в противном случае выполнение действий из п. 3.
 3. Осуществление процедуры генетического поиска при помощи модифицированных генетических операторов.
 4. Проверка на соответствие критерию остановки генетического поиска - сравнение числа выполненных итераций с минимально необходимым. В случае соответствия критерию, осуществляется переход к эволюционному поиску.
 5. Реализация оператора миграции, формирование новых подпопуляций с учетом наилучших решений, полученных при реализации генетического поиска.
 6. Выбор стратегии эволюционного поиска в зависимости от входных параметров (выбор параметров используемого оператора мутации). Одним из инструментов адаптации при реализации эволюционного поиска, является конкуренция между индивидами за право включения в следующую популяцию.
 7. Проверка на соответствие критерию эволюционного поиска - определение числа итерационных реализаций модифицированных операторов мутации. В случае соответствия критерию, осуществляется переход к моделированию генетического поиска, или вывод оптимального решения.
 8. Анализ решений, полученных в процессе выполнения генетического поиска. Осуществляется ранжирование особей популяции на основе присвоения статусов (перспективное, неперспективное, тривиальное и др.) с целью повышения эффективности бионического поиска за счет большей
-

структурированности множества альтернативных решений. Кроме того структуризация популяции упрощает динамическую корректировку направления поиска.

9. Реализация оператора миграции, формирование новой популяции с учетом наилучших решений, полученных на этапе эволюционного поиска.

10. Оценка особей, образующих популяцию, и популяции в целом.

11. Применение оператора редукции, генерация новой популяции с учетом наилучших решений.

12. Проверка на предварительную сходимость (концентрацию решений в «локальных ямах»), в случае выполнения условия, к популяции применяется блок адаптации, иначе переход к п.13.

13. Определение достижения критерия останова бионического алгоритма – выполнение заданного числа итераций и/или истечение времени отводимого для работы алгоритма.

Предлагаемый подход

При решении статической задачи оптимизации автомат адаптации можно использовать для управления процессом поиска при исполнении нечеткой команды вида «стоимость передачи потока должна быть близка к x » (рис.2).

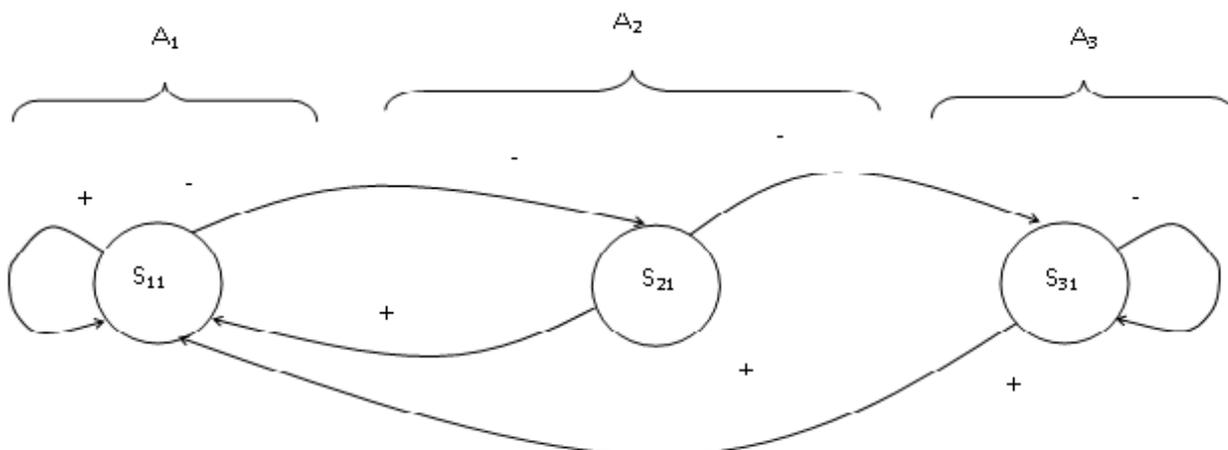


Рис. 2. – Схема переходов автомата адаптации решающего, статическую задачу

В зависимости от близости полученного промежуточного решения к x автомат адаптации принимает решение о переходе в новое состояние. Альтернатива A_1 подразумевает остановку алгоритма, а A_2 - заключается в необходимости локального пересмотра промежуточного решения, а альтернатива A_3 заключается в полном пересмотре промежуточного решения.

В случае динамической модели указанные выше матрицы приобретают вид: $\|a_{ij}(t)\|_{n,m}$, $\|c_{ij}(t)\|_{n,m}$ и $\|x_{ij}(t)\|_{n,m}$. соответственно. Рассмотрим ситуацию, когда динамичность модели подразумевает изменение пропускаемых потоков или стоимости передачи заданных потоков. В подобных ситуациях границы допустимых значений скорости и стоимости передачи данных целесообразно задавать не в четкой, а в расплывчатой форме. Например, допустимая скорость передачи данных может быть задана на основе известной формулы определения нечеткой близости $\mu_x(b)$ переменной b к заданной величине x [4,5,6]:

$$\mu_x(b) = e^{-\tau(x-b)^2},$$

где τ зависит от требуемой степени нечеткости $\mu_x(b)$ и определяется из выражения

$$\tau = -\frac{4 \ln 0,5}{\theta^2},$$

где θ расстояние между точками b_1 и b_2 для которых справедливо равенство $\mu_x(b_1) = \mu_x(b_2) = 0,5$.

Тогда критерием соответствия текущей скорости передачи данных может быть неравенство $\mu_x(b) \geq \mu_{дон}$, где $\mu_{дон}$ – минимально допустимая степень принадлежности. Если неравенство $\mu_x(b) \geq \mu_{дон}$ не соблюдается, то это является признаком необходимости пересмотра маршрутов передачи данных, чтобы снизить скачкообразность принятия решений, которая может быть обусловлена кратковременными пиковыми нагрузками, предлагается использовать автомат адаптации. Автомат адаптации, поддерживает три альтернативы A_1 , A_2 , A_3 (рис. 3). Состояние S_{11} – соответствует альтернативе

A_1 , а состояния S_{21} и S_{22} - альтернативе A_2 , состояние S_{31} - альтернативе A_3 . Альтернатива A_1 подразумевает неизменность имеющихся распределений маршрутов передачи данных; A_2 -заключается в необходимости локальной переоптимизации потоков документов, а альтернатива A_3 заключается в полной переоптимизации потоков документов.

Переход между состояниями автомата адаптации может происходить на основе нечетких результатов анализа процесса передачи данных. Пример правил выработки управляющих сигналов приведен в таблице.

Таблица.

Правила выработки управляющих сигналов

№п/п	Текущая альтернатива	Условие	Рекомендуемая альтернатива	Управляющий сигнал
1	A_1	$\mu_x(b) \geq \mu_{don}$	A_1	+
2	A_1	$\mu_x(b) \leq \mu_{don}$	A_2	-
3	A_2	$\mu_x(b) \geq \mu_{don}$	A_1	+
4	A_2	$\mu_x(b) \leq \mu_{don}$	A_3	-
5	A_3	$\mu_x(b) \geq \mu_{don}$	A_1	+
6	A_3	$\mu_x(b) \leq \mu_{don}$	A_3	-

В соответствии с правилами, приведенными в таблице, автомат адаптации изменяет своё состояние.

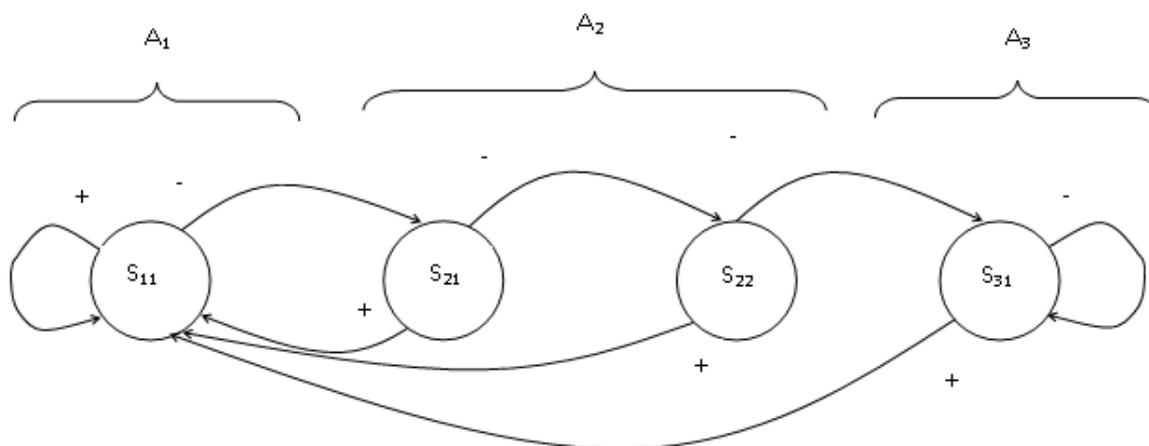


Рис. 3. – Схема переходов автомата адаптации



В случае, если необходимо провести оптимизацию процесса передачи данных, автомат адаптации переходит в состояние соответствующее альтернативе A_2 - оптимизация более быстро сходящимся алгоритмом. Если алгоритм сходится преждевременно, используются более ресурсоемкие модификации.

Заключение

В настоящее время интеллектуальные подходы активно используются для решения оптимизационных задач [7-12]. Предложен адаптивный бионический алгоритм решения задачи о потоке данных минимальной стоимости в статической и динамической постановке. При решении статической задачи алгоритм обеспечивает получение решения близкого к требуемому, а в случае динамической постановки – адаптацию процесса передачи данных под изменяющуюся модель сети. Отличительной особенностью алгоритма является использование автоматов адаптации для определения необходимости и способа модификации промежуточных решений, а также оценки необходимости изменения полученного ранее решения.

Благодарность

Работа выполнена при финансовой поддержке РФФИ – проекты 15-01-05129, 16-01-00390, 16-01-00391.

Литература

1. Чернышев Ю. О., Полуян А.Ю. Решение задачи оптимизации на основе параллельного бионического поиска // Известия ЮФУ. Технические науки. 2009. №4(93). С. 34-39.
2. Чернышев Ю.О. Остроух Е.Н. Решение задачи оптимизации алгоритмов сетевыми методами // Электроника и моделирование. 1977. № 15. С. 34-36.



3. Липский. В. Комбинаторика для программистов. М.: Мир, 1988. 200 с.
 4. Борисов А.Н., Алексеев А.В., Крумберг О.А. и др. Модели принятия решений на основе лингвистической переменной. Рига: Зинатне, 1982. 256 с.
 5. Чернышев Ю.О., Венцов Н.Н., Панасенко П.А. Алгоритм управления информационными процессами на основе нечетких и адаптивных подходов // Международный конгресс по интеллектуальным системам и информационным технологиям «IS-IT`15». М.: Физматлит, 2015, Т.1, С.281-286.
 6. Чернышев Ю.О., Венцов Н.Н., Панасенко П.А. Анализ вариантов построения обобщенных расплывчатых ограничений // Международный конгресс по интеллектуальным системам и информационным технологиям «IS-IT`15». М.: Физматлит, 2015, Т.1, С.275-281.
 7. Гинис Л.А. Развитие инструментария когнитивного моделирования для исследования сложных систем // Инженерный вестник Дона, 2013, №3 URL: ivdon.ru/ru/magazine/archive/n3y2013/1806
 8. Венцов Н.Н. Разработка алгоритма управления процессом адаптации нечетких проектных метаданных // Инженерный вестник Дона, 2012, №1 URL: ivdon.ru/ru/magazine/archive/n1y2012/630
 9. Valiant Leslie. Probably Approximately Correct: Nature's Algorithms for Learning and Prospering in a Complex World. New York: Basic Books, 2013. 208 p.
 10. Novák V. "Are fuzzy sets a reasonable tool for modeling vague phenomena?" // Fuzzy Sets and Systems 156. 2005. pp. 341–348.
 11. Earl Cox. Fuzzy Modeling and Genetic Algorithms for Data Mining and Exploration. Amsterdam: Elsevier/Morgan Kaufmann. 2005. 530 p.
 12. Остроух Е.Н., Золотарева Л.И., Бычков А.А., Долгов В.В. Векторная оптимизация перерабатывающих процессов с учетом сырьевого дефицита // Фундаментальные исследования. 2011. №12-1. С. 224-227.
-

References

1. Chernyshev Yu. O., Poluyan A.Yu. Izvestiya YuFU. Tekhnicheskie nauki. 2009. №4(93). pp. 34-39.
 2. Chernyshev Yu.O. Ostroukh E.N. Elektronika i modelirovanie. 1977. № 15. pp. 34-36.
 3. Lipskiy V. Kombinatorika dlya programmistov [Combinatorics for programmers]. M.: Mir, 1988. 200 p.
 4. Borisov A.N., Alekseev A.V., Krumberg O.A. i dr. Modeli prinyatiya resheniy na osnove lingvisticheskoy peremennoy [The decision-making model based on linguistic variable]. Riga: Zinatne, 1982. 256 p.
 5. Chernyshev Yu.O., Ventsov N.N., Panasenko P.A. Mezhdunarodnyy kongress po intellektual'nym sistemam i informatsionnym tekhnologiyam «IS-IT`15» (International Congress on intellectual systems and information technologies «IS-IT`15»). M.: Fizmatlit, 2015, T.1, pp.281-286.
 6. Chernyshev Yu.O., Ventsov N.N., Panasenko P.A. Mezhdunarodnyy kongress po intellektual'nym sistemam i informatsionnym tekhnologiyam «IS-IT`15» (International Congress on intellectual systems and information technologies «IS-IT`15»). M.: Fizmatlit, 2015, T.1, pp.275-281.
 7. Ginis L.A. Inzhenernyj vestnik Dona, 2013, №3 URL: ivdon.ru/ru/magazine/archive/n3y2013/1806
 8. Vencov N.N. Inzhenernyj vestnik Dona, 2012, №1 URL: ivdon.ru/ru/magazine/archive/n1y2012/630
 9. Valiant Leslie. Probably Approximately Correct: Nature's Algorithms for Learning and Prospering in a Complex World. New York: Basic Books, 2013. 208 p.
 10. Novák V. “Are fuzzy sets a reasonable tool for modeling vague phenomena?”. Fuzzy Sets and Systems 156. 2005. pp. 341–348.
-



11. Earl Cox. Fuzzy Modeling and Genetic Algorithms for Data Mining and Exploration. Amsterdam: Elsevier/Morgan Kaufmann. 2005. 530 p.
12. Ostroukh E.N., Zolotareva L.I., Bychkov A.A., Dolgov V.V. Fundamental'nye issledovaniya. 2011. №12-1. pp. 224-227.