

## ИССЛЕДОВАНИЕ МЕТОДОВ ЭВОЛЮЦИОННОЙ ОПТИМИЗАЦИИ В ЗАДАЧАХ ОРГАНИЗАЦИОННОГО УПРАВЛЕНИЯ ПРОИЗВОДСТВОМ

*ГВОЗДИНСКИЙ А.Н., КОЗЛОВА А.Е.*

Предлагается подход к решению проблем управления запасами, основанный на эволюционных методах, возникающих при поиске рационального плана раскроя. Демонстрируется возможность сравнительно легкой адаптации алгоритма к различным вариантам постановки задачи. В качестве объектов исследования рассматриваются системы интеллектуального анализа данных и объекты производственной деятельности предприятий и фирм различного назначения. Рассматривается задача управления запасами в системах производственного планирования на примере задачи оптимального раскроя ресурсов.

### 1. Введение

В данной работе приводится формальная постановка задачи, на решение которой изначально был направлен первый генетический алгоритм (ГА), и общая схема работы самого алгоритма. Описываются составляющие ГА и их назначение. Для решения конкретной задачи разработчик, исследователь может сконструировать свою конкретную схему работы генетического алгоритма из его составляющих. При этом разработчик руководствуется особенностями предметной области, формализации задачи, структурой используемых данных, возможно, даже результатами тестирования других схем ГА. Разумеется, в строении всех таких алгоритмов есть общие элементы и последовательность действий, которые являются основой генетических алгоритмов и отличают их от многих других. Данная статья посвящена опыту конструирования конкретных вариантов ГА.

### 2. Анализ методов эволюционной оптимизации

Средства эволюционной оптимизации включают следующие методы: нейронные сети, деревья решений, генетические алгоритмы, а также их комбинации [1].

Нейронные сети относят к классу нелинейных адаптивных систем, строением они условно напоминают нервную ткань из нейронов. Это набор связанных друг с другом узлов, получающих входные данные,

которые осуществляют их обработку и вырабатывают на выходе определенный результат. На узлы нижнего слоя подаются значения входных параметров, на их основе производятся вычисления, необходимые для принятия решений, прогнозирования развития ситуации и т.д. Эти значения рассматривают как сигналы, которые передаются в вышележащий слой, усиливаясь или ослабляясь в зависимости от числовых значений (весов), приписываемых межнейронным связям. На выходе нейрона самого верхнего слоя вырабатывается значение, которое рассматривается как ответ, реакция всей сети на введенные начальные значения. Так как каждый элемент нейронной сети частично изолирован от своих соседей, у таких алгоритмов имеется возможность для распараллеливания вычислений. Нейроны сети обрабатывают входные данные, для которых известны и значения входных параметров, и правильные ответы на них. Обучение состоит в подборе весов межнейронных связей, которые обеспечивают наибольшую близость ответов сети к известным правильным ответам. После обучения на имеющихся данных сеть готова к работе и может быть использована для построения прогнозов поведения объекта в будущем. Нейронные сети используются для решения задач прогнозирования, классификации или управления.

Достоинство – сети могут аппроксимировать любую непрерывную функцию, нет необходимости заранее принимать какие-либо предположения относительно модели. Исследуемые данные могут быть неполными или зашумленными. Удобны при работе с нелинейными зависимостями.

Недостаток – необходимость иметь большой объем обучающей выборки. Окончательное решение зависит от начальных установок сети. Данные должны быть обязательно преобразованы к числовому виду. Полученная модель не объясняет обнаруженные знания.

Деревья решений используют разбиение данных на группы на основе значений переменных. В результате получается иерархическая структура операторов «Если..., то...», которая имеет вид дерева. Для классификации объекта или ситуации нужно ответить на вопросы, стоящие в узлах этого дерева, начиная от его корня. Если ответ положительный, переходят к правому узлу следующего уровня, если отрицательный – к левому узлу и т.д. Заканчивая ответы, доходят до одного из конечных узлов, где указывается, к какому

классу надо отнести рассматриваемый объект. Деревья решений предназначены для решения задач классификации и поэтому весьма ограничено применяются в области финансов и бизнеса.

Достоинство – простое и понятное представление признаков для пользователей. В качестве целевой переменной используются как измеряемые, так и неизмеряемые признаки – это расширяет область применения метода.

Недостаток – проблема значимости. Данные могут разбиваться на множество частных случаев, возникает “кустистость” дерева, которое не может давать статистически обоснованных ответов. Полезные результаты получают только в случае независимых признаков.

Генетические алгоритмы на данный момент являются наиболее известным представителем эволюционных методов оптимизации. Они содержат все основные генетические операции. ГА получены в процессе обобщения и имитации в искусственных системах свойств живой природы:

- приспособляемость к изменениям среды;
- естественный отбор;
- наследование потомками наиболее «ценных» свойств родителей.

С помощью ГА можно улучшать работу поисковых систем, которые требуют обработки больших массивов информации; оптимизировать работу нефтяных трубопроводов; распределять инструменты в металлообрабатывающих цехах; осуществлять оптимизацию профилей балок в строительстве. Решение задач комбинаторной оптимизации является одной из основных областей применения ГА. Генетические алгоритмы имитируют процесс естественного отбора в природе. Для нахождения более оптимального критерия все решения описываются набором чисел или величин нечисловой природы. Поиск оптимального решения похож на эволюцию популяции индивидов, которые представлены наборами их хромосом. В этой эволюции действуют три механизма:

- отбор сильнейших наборов хромосом, которым соответствуют наиболее оптимальные решения;
- скрещивание – получение новых индивидов при помощи смешивания хромосомных наборов отобранных индивидов;
- мутации – преобразование хромосомы, случайное изменение одного или несколько генов (чаще – одного).

Наиболее популярное приложение генетического алгоритма – оптимизация многопараметрических функций. Реальные задачи формируются как поиск оптимального значения сложной функции, зависящей от некоторых  $n$  – входных параметров. Сила ГА – в способности манипулировать одновременно многими параметрами. В одних случаях получено точное

решение функции, в других – решением считается любое значение, лучшее некоторой заданной величины.

Чтобы реализовать генетический алгоритм, надо сначала выбрать структуру для представления этих решений. Структура данных ГА состоит из одной или большего количества хромосом. После проведенных в ходе применения ГА операций осуществляем декодирование и получаем подмножество значений, которые соответствуют субоптимальному решению задачи. Генетический алгоритм оптимизации является множественно-вероятностным, т.е. он позволяет находить множество значений, которые примерно соответствуют искомому условию. Это существенно для решения задач с неявно выраженными максимумами или минимумами.

Существует множество модификаций генетического алгоритма, которые отличаются методами отбора, скрещивания, хромосомной кодировки. Традиционный (основной) ГА работает с двоичной строкой определенной длины, использует следующие свойства:

- на каждом поколении реализуется отбор пропорционально функции приспособленности;
- одноточечный кроссовер (скрещивание);
- мутация.

Как метод оптимизации ГА обладает внутренним параллелизмом: различные существенные комбинации генов отыскиваются параллельным образом, одновременно для всех комбинаций. Чем меньше комбинация, тем легче ее определить. В связи с тем, что алгоритм в процессе поиска использует некоторую кодировку значений, а не сами значения, он эффективно может применяться для решения задач дискретной оптимизации, которые могут быть определены как на числовых, так и на конечных множествах произвольной природы. Так как в качестве информации об оптимизируемой функции для работы алгоритма используются лишь значения в рассматриваемых точках пространства поиска, то данный алгоритм применим к широкому классу функций. Генетический алгоритм успешно справляется с поставленной задачей даже там, где не существует общеизвестных алгоритмов решения или высока степень априорной неопределенности.

Достоинство – метод удобен для решения различных задач комбинаторики и оптимизации, предпочтителен больше как инструмент научного исследования.

Недостаток – возможность эффективно сформулировать задачу, определить критерий отбора хромосом, и сама процедура отбора является эвристической и под силу только специалисту. Постановка задачи в терминах не дает возможности проанализировать статистическую значимость получаемого с их помощью решения.

### 3. Формулировка задачи и построение математической модели

Для построения математической модели за основу возьмем один из методов интеллектуального анализа данных – генетические алгоритмы, и на их базе будем решать поставленную задачу.

*Целью работы* является исследование деятельности предприятия, составление математической модели задачи в классе экстремальных задач и определение на основе решения оптимальной стратегии работы фирмы на определенный срок, а также анализ возможных колебаний затрат и цен реализации, которые не приведут к изменению стратегии.

Формально, *генетический алгоритм* – это любой алгоритм поиска приближенного решения поставленной задачи, построенный по *схеме генетического алгоритма*.

*Шаг 0.* Создать случайную начальную популяцию  $\{S_k(0)\}$ .

*Шаг 1.* Вычислить приспособленность  $f(S_k)$  каждой особи  $(S_k)$  популяции  $\{S_k(t)\}$ .

*Шаг 2.* Производя отбор особей из  $\{S_k(t)\}$  в соответствии с их приспособленностями  $f(S_k)$  и применяя генетические операторы (скрещивания и мутации) к отобраным особям для получения потомства, сформировать популяцию следующего поколения  $\{S_k(t+1)\}$ .

*Шаг 3.* Повторять шаги 1, 2 для  $t = 0, 1, 2, \dots$  до тех пор, пока не выполнится некоторое условие окончания эволюционного поиска (прекращается рост максимальной приспособленности в популяции, число поколений  $t$  достигает заданного предела).

Возможны различные варианты генетического алгоритма, которые отличаются по схемам отбора особей из текущего поколения в новое, операторам скрещивания и мутации хромосом особей, по форме представления хромосом. Традиционный вид генетического алгоритма базируется на следующей частной схеме:

– цепочки символов в хромосомах бинарные (символы  $S_{kj}$  принимают значения 0 либо 1), длина цепочек постоянна ( $N = \text{const}$ );

– метод отбора – пропорционально-вероятностный;

– скрещивания производятся по схеме одноточечного.

Рассмотрим распределение ресурсов на примере решения задачи оптимального раскроя [2].

На заготовочный участок радиоцеха поступили листы фольгированного гетинакса размером  $1_1 \times 1_2$  мм. Его необходимо разрезать на заготовки для печатных плат:  $\Pi_1$  размером  $1_3 \times 1_4$  мм,  $\Pi_2$  –  $1_5 \times 1_6$  мм,  $\Pi_3$  –  $1_7 \times 1_8$  мм,  $\Pi_4$

–  $1_9 \times 1_{10}$  мм, необходимость в которых  $\Pi_1, \Pi_2, \Pi_3, \Pi_4$  единиц соответственно.

Построить модель, на ее основании составить экстремальную задачу, решение которой даст минимальное количество разрезанных выходных листов. Найти решение задачи в условиях минимизации отходов раскроя. Предусмотреть возможность максимизации количества комплектов печатных плат при условии выполнения комплектности  $\Pi_1:\Pi_2:\Pi_3:\Pi_4 = k_1:k_2:k_3:k_4$  (таблица).

Параметры	Значение
$1_1$	136
$1_2$	158
$1_3$	92
$1_4$	36
$1_5$	51
$1_6$	88
$1_7$	46
$1_8$	40
$1_9$	50
$1_{10}$	50
$\Pi_1$	270
$\Pi_2$	200
$\Pi_3$	310
$\Pi_4$	90
Компетентность	3:2:1:5

Исходные данные:  $1_1 \times 1_2, \Pi_1, \Pi_2, \Pi_3, \Pi_4, k_1, k_2, k_3, k_4$ .

Используемые переменные:  $\Pi_i = a_i$  – необходимое количество заготовок  $i$ -го типа;  $C_j$  – отход при раскрое единиц исходного материала  $j$ -м способом;  $j = \overline{1, m}$  – количество вариантов раскроя;  $a_{ij}$  – количество заготовок  $i$ -го типа ( $i = \overline{1, 4}$ ), полученных при раскрое единицы исходного материала  $j$ -м вариантом;

$i$  – вид заготовки;  $q$  – количество изделий ( $\Pi_1, \Pi_2, \Pi_3, \Pi_4$ );  $a$  – количество разрезанных листов;  $K$  – количество типов листов, поступающих на раскрой ( $K = 1$ );

$1_i$  – компетентность-количество заготовок  $i$ -го вида в одном комплекте;  $x_j$  – количество единиц исходного материала, который раскраивается  $j$ -м вариантом;

### 4. Формализация исследуемой задачи

Сформируем ограничения для данной задачи:

$$\sum_{j=1}^m x_j = a, \quad (1)$$

$$x_j \geq 0, j = \overline{1, m}, \quad (2)$$

$$\sum_{j=1}^m a_{ij} x_j = \Pi_i. \quad (3)$$

Функция цели имеет вид:

$$F(x) = \sum_{j=1}^m x_j \rightarrow \min, \quad (4)$$

$$F(x) = a \rightarrow \min. \quad (5)$$

Функция цели имеет вид:

$$\sum_{j=1}^n C_j x_j \rightarrow \min. \quad (6)$$

Необходимо предусмотреть возможность максимизации количества комплектов печатных плат при условии выполнения комплектности  $\Pi_1:\Pi_2:\Pi_3:\Pi_4 = k_1:k_2:k_3:k_4$ .

Функция цели имеет вид:

$$F(x) = X \rightarrow \max, \quad (7)$$

где  $X = \frac{1}{\Pi_1 k_1} \left( \sum_{j=1}^m a_{j1} x_j \right)$ .

А ограничения (3) примут следующий вид:

$$\sum_{j=1}^p a_{j1} x_j = \Pi_1 k_1 X, \quad \sum_{j=1}^p a_{j2} x_j = \Pi_2 k_2 X, \quad (8)$$

$$\sum_{j=1}^p a_{j3} x_j = \Pi_3 k_3 X, \quad \sum_{j=1}^p a_{j4} x_j = \Pi_4 k_4 X.$$

## 5. Алгоритмы решения исследуемых задач

Для начала рассмотрим решение данной задачи на основе методов линейного программирования [3].

Алгоритм симплекс-метода разделяют на два этапа нахождения: начального опорного плана задачи и оптимального плана, или установление факта неограниченности целевой функции на множестве планов задачи.

Алгоритм симплекс-метода применяется лишь для канонической формы задач, поэтому задачу в других формах нужно изображать также в канонической форме.

Общая постановка задачи линейного программирования – определить величины  $x_j \geq 0, j = \overline{1, n}$ , которые максимизируют линейную форму

$$Z = \sum_{j=1}^n C_j x_j \rightarrow \max \quad (9)$$

при условиях существования ограничений вида

$$AX \leq B, \quad (10)$$

$$A = \| a_{ij} \|_{mn}, X = \| x_1, x_2, \dots, x_n \|, B = \| b_1, b_2, \dots, b_m \|.$$

Симплекс-метод применяется лишь для канонической формы задачи линейного программирования, поэтому, чтобы начать решение с помощью симплекс-метода, необходимо превратить систему неравенств в эквивалентную систему уравнений путем введения новых неотъемлемых переменных  $x_{n+1}, x_{n+2}, \dots, x_{n+m}$ . Вследствие этого система приобретает вид:

$$\begin{aligned} a_{11} x_1 + a_{12} x_2 + \dots + a_{1n} x_n + x_{n+1} &= b_1, \\ a_{21} x_1 + a_{22} x_2 + \dots + a_{2n} x_n + x_{n+2} &= b_2, \\ \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots & \dots \dots \dots \end{aligned} \quad (11)$$

$$a_{m1} x_1 + a_{m2} x_2 + \dots + a_{mn} x_n + x_{n+m} = b_m.$$

Анализ методов традиционной оптимизации показывает, что имеется ряд недостатков, связанных с ограничением на размерность задач. Поэтому предлагается использовать генетические алгоритмы для решения совместной задачи.

Требуется подобрать алгоритм, с помощью которого можно эффективно решить задачу. Был выбран генетический алгоритм по следующим причинам:

*во-первых*, задача раскроя любого вида является ОТ-полной;

*во-вторых*, время получения решения для заказчиков предполагается порядка десятки секунд для размерности 100, это достаточно высокие требования ко времени;

Для реализации ГА особью предлагается считать любое размещение заданных предметов на полосе, удовлетворяющее условию задачи. Пусть все предметы некоторым образом пронумерованы. Тогда пусть хромосома — это перечисление всех номеров предметов в некотором порядке, отражающем их размещение. Физический внешний вид особи, конкретные координаты размещения предметов на полосе можно получить, выложив предметы на полосу в порядке их перечисления в хромосоме по правилам переработки — декодирования. Эти правила называются процедурой декодирования, или декодером. Различные декодеры могут работать по разной схеме, но суть одна — выложить предметы на полосу согласно некоторому правилу в порядке, указанном в хромосоме. Правила являются эвристическими и существенно строятся так, чтобы размещение было по возможности плотным.

В качестве критерия оптимизации, критерия оценки пригодности особи в поставленных задачах рассматривается длина полосы, занятая размещением предметов.

Пустоты учитываются и нумеруются, начиная с числа, на единицу большего количества предметов, но со знаком минус. Декодер, работающий с такой структурой данных, называется блочным. Размещение представлено в виде совокупности кортежей (блоков):  $(1., 2., \dots)x_j$ , где  $j = 1, 2, \dots$

А именно:  $(1; 2) x_1, (3; 4; -9; 2) x_2, (3; 5; 6) x_3, (7; -10; 5; 6) x_4, (7; 8; 6) x_5, (-11) x_6$ . Длина занятой части полосы  $M = \wedge x$ . Одна из особенностей блочного декодера в том, что размещая очередной предмет, он сначала проверяет все образовавшиеся ранее пустоты, подойдут ли они по размеру, для достижения большей плотности размещения (рис. 1).

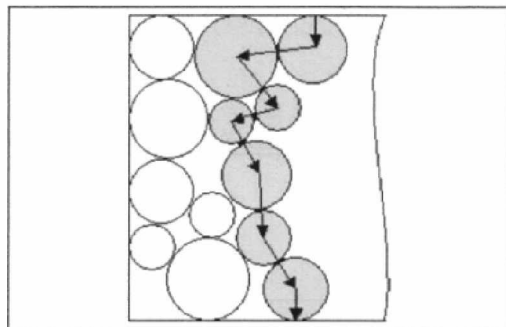


Рис. 1. Схема края размещения

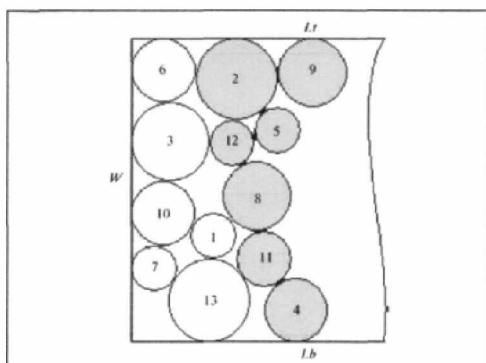


Рис. 2. Формальная запись края размещения

Для лучшего описания понятия края упаковки можно провести физическую аналогию. Пусть торец полосы  $W$  – дно двумерного стакана, нальем в него воду. Все хотя бы частично намочившие предметы – это край полосы (рис. 2).

При помощи специальной структуры данных краевой декодер хранит информацию о крае текущего размещения и располагает каждый следующий предмет вплотную к краю, минимизируя общую занятую предметами длину полосы.

Для решения задачи предлагается модификация генетического алгоритма, использующая процедуру декодирования как параметр. Сам генетический алгоритм (т.е. схема его работы, смена поколений), структура хромосомы, операторы мутации, скрещивания, селекции не чувствительны к геометрии размещаемых предметов. Например, оператор мутации случайным образом переставляет два номера предметов в хромосоме. Оператор скрещивания двух хромосом

порождает двух потомков. Начальный участок дочерней хромосомы совпадает со случайным участком хромосомы одного родителя, а конец состоит из оставшихся предметов, перечисленных в том порядке, в котором они следуют в хромосоме другого из родителей.

Например, оператор скрещивания (для случайных параметров  $p = 3, q = 4$ ) для родителей  $S_{p1} = (1, 2, 3, 4, 5, 6, 7, 8, 9)$  и  $S_{p1} = (4, 7, 9, 1, 2, 5, 3, 8, 6)$  порождает потомка  $S_{c1} = (3, 4, 5, 6, 7, 9, 2, 8)$  и для тех же родителей (для удобства понимания по-другому записанных)  $S_{p2} = (4, 7, 9, 1, 5, 3, 8, 6)$  и  $S_{p2} = (1, 2, 3, 4, 5, 6, 7, 8, 9)$  – потомка  $S_{c2} = (9, 1, 2, 5, 3, 4, 6, 7, 8)$ . Оператор мутации (для случайных параметров  $a = 2, b = 7$ ) исходную особь  $S = (9, 8, 7, 6, 5, 4, 3, 2, 1)$  преобразует в особь вида  $S_m = (9, 3, 7, 6, 5, 4, 8, 2, 1)$ .

Генетический алгоритм показал, что оценочная полиномиальная сложность алгоритма и относительно невысокие времена работы программы не зависят от типа задачи. В то же время оценочная сложность блочного декодера пропорциональна квадрату размерности задачи (числу предметов), а краевого — линейно зависит от размерности.

Разработанный алгоритм ориентирован на расширение и модификацию. Смена декодера позволяет решать задачу раскроя полубесконечной полосы как на прямоугольные, так и на круглые предметы. Предположительно, если воспользоваться таким эвристическим приемом, как годограф Хэкманна, в качестве декодера, то тот же алгоритм возможно использовать для раскроя полубесконечной полосы на предметы нерегулярной формы.

Обе изначально поставленные задачи раскроя предполагают размещение предметов не на полубесконечной полосе, как обсуждалось при формализации задач, а на наборе прямоугольных листов. Данный алгоритм был модифицирован и для решения задач раскроя в таких, действительно прикладных, а не абстрактных постановках.

В реальном производстве, например, металлургическом, все листы имеют одинаковую ширину. Это выгодное обстоятельство дает возможность не рассматривать набор отдельных листов, а условно разбить на них непрерывную полосу. Предложено оставить в математической постановке задачи полубесконечную полосу, но добавить ограничение: предметы не должны пересекать границы между листами.

При модификации алгоритма остались неизменными все генетические операторы и сама схема работы генетического алгоритма.

## 6. Выводы

Исследованы проблемы и существующие методы управления запасами на предприятиях, разработана модель объекта управления и процессов управления. Выявлены наиболее значимые, с точки зрения повышения эффективности, процессы принятия решений в

системе управления запасами. Исследованы методики поддержки принятия решений в процессах управления поступлением, хранением ресурсов, основанных на предложенной модели расчета затрат. Разработана методика принятия решений в целях оптимизации системы управления запасами материально-технических ресурсов, используемых на предприятиях. Проведен анализ эффективности разработанной методики управления запасами;

*Научная новизна* исследования заключается в обосновании экономического механизма управления запасами материально-технических ресурсов и разработке методического обеспечения процесса принятия решений по оптимизации объемов запасов.

Основными результатами исследования являются: обоснование места, роли и рациональной структуры запасов в системе планирования сбытовой деятельности предприятия в современных условиях; разработка концептуальных подходов к решению оптимизационных задач в области управления запасами материально-технических ресурсов, использующих инструментальный анализ и процесса принятия решений; определение факторов стратегии управления запасами, основанных на различных параметрах системы закупок, таких как периодичность заказа, постоянная партия объема пополнения заказа, уровень восполнения запаса. Предложена методика анализа состояния запасов, основанная на анализе изменения структуры и объема затрат предприятия, позволяющая контролировать состояние запасов в реальном режиме времени.

*Практическая значимость* определена возможностью применения разработанной методики в области управления запасами на малых предприятиях, использующих в производственном процессе незначительное количество материальных ресурсов, как по номенклатуре, так и по объему. Разработанные теоретические и методические положения доведены до практических моделей и алгоритмов, в результате чего появляется возможность дать количественное обоснование размеров партий заказов, закупок и продаж, повышается достоверность расчетов.

**Литература:** 1. *Гвоздинский А.Н.* Методы аналитической обработки информации / А.Н. Гвоздинский, Е.Г. Клишко // Радиоэлектроника и информатика. 2000. №4. С.111-112. 2. *Гвоздинский А.Н.* Принципы и методы оптимизации в современных организационных системах управления // X: Бионика интеллекта, 2013. №2(81). С.66-70. 3. *Гвоздинский А.М.* Методи оптимізації в системах прийняття рішень // X: ХНУРЕ, 2006. С.327.

Поступила в редколлегию 23.11.2013

**Рецензент:** д-р техн. наук, проф. Куземин А.Я.

**Гвоздинский Анатолий Николаевич**, канд. техн. наук, профессор кафедры искусственного интеллекта ХНУРЭ. Научные интересы: оптимизация процедур принятия решений в сложных системах управления. Адрес: Украина, 61166, Харьков, ул. Академика Ляпунова, 7, кв.9, тел. 702-38-23.

**Козлова Алина Евгеньевна**, студентка группы КН-10-4 кафедры искусственного интеллекта ХНУРЭ. Научные интересы: методы принятия решения в системах искусственного интеллекта. Адрес: Украина, 61000, Харьков, пр. Героев Сталинграда, кв.47, тел. (050)-61-30-071.