УДК 519.168+004.021

РЕШЕНИЕ ЗАДАЧИ РАЗМЕЩЕНИЯ НАГРУЖЕННЫХ ОТРЕЗКОВ НА МАССИВЕ ПЕРЕСТАНОВОК «ПУЗЫРЕК»

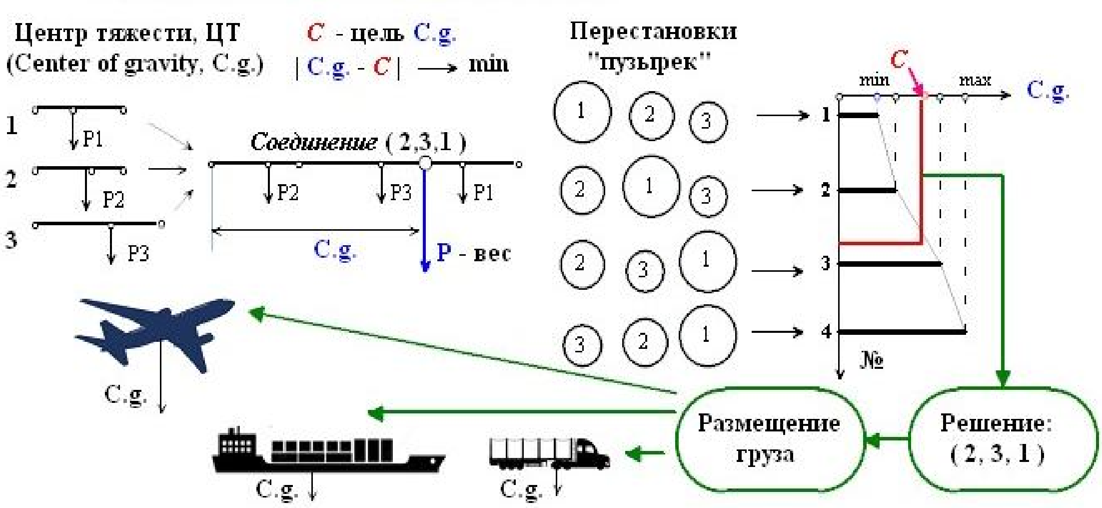
Статья поступила в редакцию 10.01.2019, в окончательном варианте – 20.01.2019.

Дворянкин Александр Михайлович, Волгоградский государственный технический университет, 400005, Российская Федерация, г. Волгоград, пр. им. Ленина, 28,
доктор технических наук, профессор, e-mail: dvam1@mail.ru

Приведено описание основных используемых понятий: нагруженный отрезок, соединение отрезков, определение порядка на множестве отрезков, формула для вычисления центра тяжести соединения отрезков. Сформулирована задача размещения множества отрезков, нагруженных сосредоточенной силой. Предложено использование в качестве критерия точности решения показателя «отклонение центра тяжести от целевого положения». Задача является *NP* полной. Алгоритм полного перебора для ее решения имеет временную сложность $O(n!)$, где n – число размещаемых отрезков. При большом числе отрезков получение оптимального решения путем сплошного перебора вариантов требует очень высоких вычислительных затрат. Поэтому поиск эффективных, приближенных алгоритмов является актуальной задачей. Приводится определение массива перестановок «пузырек». Предложен алгоритм формирования транспозиций, с помощью которых можно формировать перестановки этого массива. Доказаны следующие теоремы: о неубывании положения центра тяжести соединения отрезков на этих перестановках; о вычислении приращения положения центра тяжести на соседних перестановках. На основе этих теорем построен алгоритм приближенного решения задачи размещения отрезков, временная сложность которого $O(n^2)$. Дана формула, определяющая для заданных отрезков максмально возможное отклонение. Приведены примеры с результатами работы алгоритма. Результаты работы могут быть использованы для планирования оптимального размещения груза на транспорном средстве.

Ключевые слова: нагруженный отрезок, соединение отрезков, размещение отрезков, центр тяжести, перестановка, транспозиция, массив «пузырек»

Графическая аннотация (graphical annotation)



SOLUTION OF THE PROBLEM OF PLACEMENT LADED SEGMENTS ON ARRAY PERMUTATIONS "BUBBLE"

The article was received by editorial board 10.01.2019, in the final version – 20.01.2019

Dvoryankin Alexandr M., Volgograd State Technical University, 28 Lenin Ave., Volgograd, 400005, Russian Federation,
Doct. Sci. (Engineering), Professor, e-mail: dvam1@mail.ru

The article provides a description of the basic concepts: a loaded segment, the connection of segments, the definition of order on a set of segments, a formula for calculating the center of gravity of the connection of segments. The problem of locating a set of segments loaded with a concentrated force is formulated. The criterion of optimal placement is the minimum deviation of the center of gravity from the given one. The task is *NP* complete. The brute-force algorithm for solving it has the time complexity $O(n!)$, Where n is the number of segments to be placed. Therefore, the search for efficient, approximate algorithms is an urgent task. The definition of the “bubble” permutation array is given. An algorithm is proposed for generating transpositions, with the help of which it is possible to form permutations of this array. The theorem on non-decreasing of the position of the center of gravity of the connection of segments on these permutations is proved. A theorem on calculating the increment of the center of gravity position on adjacent permutations is proved. On the basis of these theorems, an algorithm is constructed for the approximate solution of the problem of allocating segments, the time complexity of which is $O(n^2)$. The concept of accuracy of the solution is given – deviation of the center of gravity from the target position. A formula is given that determines the maximum possible deviation for given segments. Examples of the algorithm are given. The results of the work can be used to plan the optimal placement of cargo on the vehicle.

Key words: loaded segment, connection of segments, placement of segments, center of gravity, permutation, transposition, "bubble" array

Введение. Имеются различные постановки и алгоритмы решения задач одномерной оптимальной упаковки (размещения): «задачи о ранце» [1], упаковка одномерных контейнеров [6], представление перестановками решений одномерной задачи упаковки [4], оптимальная загрузка летательного аппарата [8, 10]. Задача одномерной упаковки является задачей комбинаторной оптимизации и относится к классу NP-полных [2]. Для ее решения разрабатываются различные алгоритмы, позволяющие получать результаты за приемлемое время и с заданной точностью [5]. Однако в настоящее время не существует универсального алгоритма, способного одинаково эффективно решать все тестовые задачи [5]. Поэтому разработка новых вычислительно эффективных алгоритмов является актуальной и важной проблемой. Этой цели и посвящена настоящая работа.

В статье рассматривается модель и задача размещения отрезков, нагруженных сосредоточенной силой [3]. В качестве критерия оптимального размещения было выбрано «минимальное отклонение центра тяжести от заданного положения». Рассматриваемые задачи возникают, в частности, в связи с необходимостью оптимизации размещения груза на транспортном средстве, для которого положение центра тяжести является важным. Предложены алгоритмы решения задачи на массиве перестановок «пузырек» [7]. Временная сложность алгоритмов $O(n^2)$, что на два порядка лучше алгоритма, описанного в работе [3]. Приведены результаты решения задач на примерах.

Описание задачи на перестановках. Приведем обозначения и формулировку задачи, предложенные в [3]. Горизонтальный нагруженный отрезок обозначается так: $A = (a, p, b)$, где a – расстояние от левого края отрезка до точки приложения силы p, b– расстояние, соответственно, до правого края отрезка. Сила p интерпретируется как вес отрезка, а точка приложения силы как центр тяжести отрезка. Компоненты отрезка A обозначаются также следующим образом: $a = a(A)$, $p = p(a)$, $b = b(A)$.Для обозначения различных отрезков используются нижние индексы. Соединение $A_{ij} = A_iA_j$ двух различных отрезков $A_i = (a_i, p_i, b_i)$ и $A_i = (a_j, p_j, b_j)$ определяется следующим образом:

$$a_{ij} = a(A_{ij}) = [p_i a_i + p_j(a_i + b_i + a_j)]/p_{ij}, \tag{1}$$

$$p_{ij} = p_i + p_j, \tag{2}$$

$$b_{ij} = b(A_{ij}) = [p_j b_j + p_i(a_j + b_j + b_i)]/p_{ij}. \tag{3}$$

На отрезках определяется отношение порядка следующим образом:

$$A_x \le A_y \Leftrightarrow \gamma_x \ge \gamma_y , \tag{4}$$

где $A_x = (a_x, p_x, b_x)$, $A_y = (a_y, p_y, b_y)$, $\gamma_x = p_x / (a_x + b_x)$, $\gamma_y = p_y / (a_y + b_y)$. Здесь γ_x, γ_y – удельные веса отрезков A_x и A_y. В работе [1] доказано (лемма 5), что для любых отрезков A_x, A_y, A_z, A_uсправедливо:

$$A_y \le A_z \Rightarrow a(A_xA_yA_zA_u) \le a(A_xA_zA_yA_u). \tag{5}$$

Для компоненты $a(A)$ соединения $A = A_1A_2...A_n$ отрезков будем использовать формулу (6) [3]:

$$a(A) = \left[\sum_{k=1}^{n} p_k\left(a_k + \sum_{i=1}^{k-1}(a_i + b_i)\right)\right] / p(A), \text{где } p(A) = \sum_{i=1}^{n} p_i . \tag{6}$$

Задача размещения отрезков формулируется следующим образом.

Дано: n отрезков A_1, A_2, …, A_n, и некоторое число c – заданное (целевое) положение центра тяжести для результата соединения этих отрезков. Пусть J_n – множество перестановок чисел $\{1, 2,…,n\}$. Любой перестановке $w = (i_1,i_2,…,i_n) \in J_n$ можно сопоставить соединение отрезков: $Aw = Ai_1Ai_2…Ai_n$. Это означает размещение отрезков в порядке перестановки. Первая компонента такого соединения отрезков есть функция $a(Aw) = a(Ai_1Ai_2…Ai_n)$ – расстояние от левого края соединенного отрезка до его центра тяжести. Для заданного множества n отрезков и числа c определяется функция δ:

$$\delta(i_1, i_2, \dots, i_n, c) = |a(Ai_1Ai_2 \dots Ai_n) - c|, \text{где } (i_1, i_2, \dots, i_n) \in J_n. \tag{7}$$

Найти: перестановку $(i_1^0, i_2^0, \dots, i_n^0) \in J_n$, на которой достигается минимум функции δ:

$$\delta(i_1^0, i_2^0, \dots, i_n^0, c) = \min_{(i_1,i_2,\dots,i_n)\in J_n} \delta(i_1, i_2, \dots, i_n, c). \tag{8}$$

Перестановка $(i_1^0, i_2^0, \dots, i_n^0) \in J_n$ определяет оптимальное размещение отрезков с минимальным отклонением центра тяжести от заданного положения c.

Алгоритм формирования массива перестановок «пузырек». Массив перестановок получил название «пузырек» по аналогии с алгоритмом «пузырьковой сортировки» [7]. К отрезкам предъявляется требование упорядочивания:

$$A_1 \le A_2 \le \cdots \le A_n. \quad (9)$$

По определению порядка на отрезках (4) это означает неубывание удельных весов отрезков. Если исходные данные не соответствуют этому условию, то необходимо произвести сортировку отрезков по неубыванию удельного веса и присвоить отрезкам новые номера. При этом можно использовать алгоритм «быстрой сортировки» [7], сложность которого $O(n\log_2 n)$.

Пусть имеется перестановка $I = (i_1,i_2,\ldots,i_j,i_{j+1},\ldots,i_n)$. Действие транспозиции $(j, j+1)$ на перестановку I означает перемену местами элементов i_j и i_{j+1} , т.е. $I\text{o}(j,j+1) = (i_1,i_2,\ldots,i_{j+1},i_j,\ldots,i_n)$ [6].

Алгоритм_1 формирования массива транспозиций «пузырек». Алгоритм основан на последовательном применении транспозиций двух соседних элементов и формирует массив транспозиций T и массив пар переставляемых элементов E.

Шаг 0. Задать число размещаемых отрезков n (размерность задачи).

Шаг 1. Положить $m = n(n-1)/2 +1$ – размерность массивов T и E. Определить массивы:

1) Начальная перестановка $Q = \{1,2,\ldots,n\}$, Qx – рабочая перестановка;

2) Масссивы $T[m]$ и $E[m]$, где T – пары индексов элементов (транспозиций) перестановки Qx; Е – пары переставляемых элементов соответствующей транспозиции. Если транспозиция $T[k] = (i, j)$, то

$E[k] = (Qx[i], Qx[j])$ для $k = 1,\ldots,m$; $E[k,1] = Qx[i]$, $E[k,2] = Qx[j]$ и $T[k,1] = i$, $T[k,2] = j$.

3) Инициализация массивов: $Qx = Q$, $k = 1$, $T[1] = (1,1)$, $E[1] = (1,1)$.

Шаг 2. Выполнить двойной цикл:

```
FOR i =1TO n-1
   FOR j =1 TO n -i
      k := k + 1
      T[k]     := (j,j+1)
      E[k]     := (Qx[j], Qx[j+1])
      // Транспозиция: Qx = QxoT[k]
      Qx[j+1] := E[k,1]
      Qx[j]   := E[k,2]
   NEXT j
NEXT i
```

Шаг 3. Результат: массив транспозиций T и пар элементов E. Конец.

Замечание 1. Временная сложность алгоритма_1 $O(n^2)$.

Замечание 2. Так как выполнено условие (9), то $E[k,1]<E[k,2]$ для всех $k = 1,2,\ldots,m$.

Замечание 3. Перестановку Qx на итерации k обозначим $Qx(k)$. Эту перестановку можно определить последовательным применением k транспозиций из массива T:

$$Qx(k) = Q\text{o}T[1]\text{o}T[2]\text{o}\ldots\text{o}T[k]. \quad (10)$$

Здесь порядок применения транспозиций определяется как «слева-направо».

Определение. Множество $\{Qx(k)\}_{k=\overline{1,m}}$ назовем множеством перестановок «пузырек». Отметим, что сложность получения перестановки $Qx(k)$ для заданного k равна $O(n^2)$.

Пример 1. Результат работы алгоритма_1 для $n = 5$ приведен в таблице 1.

Таблица 1 – Массивы T и Е транспозиций «пузырек»

k	***T*[*k*]**	***E*[*k*]**
1	(1, 1)	(1, 1)
2	(1, 2)	(1, 2)
3	(2, 3)	(1, 3)
4	(3, 4)	(1, 4)
5	(4, 5)	(1, 5)
6	(1, 2)	(2, 3)
7	(2, 3)	(2, 4)
8	(3, 4)	(2, 5)
9	(1, 2)	(3, 4)
10	(2, 3)	(3, 5)
11	(1, 2)	(4, 5)

Обозначим через $A_{Qx(k)}$ соединение отрезков на перестановке $Qx(k)$. Для примера 1 при $k = 4$:

$$Qx(4) = Q\text{o}(1,1)\text{o}(1,2)\text{o}(2,3)\text{o}(3,4) = (2,3,4,1,5), \quad (11)$$

$$A_{Qx(k)} = A_2A_3A_4A_1A_5. \quad (12)$$

Теорема 1. Неубывание функции $a(A)$ – положения центра тяжести (см. формулу (6)) на массиве «пузырек». Пусть имеются n отрезков $\{A_i\}_{i=\overline{1,n}}$, удовлетворяющих условию (9). Сформированы массивы транспозиций «пузырек» T, E и m – мощность этих массивов. Тогда для $k = \overline{1, m-1}$ справедливо неравенство:

$$a\left(A_{Qx(k)}\right) \leq a\left(A_{Qx(k+1)}\right) \quad (13)$$

Доказательство.

Пусть

$$Qx(k) = (i_1, \dots, i_{r-1}, i_r, i_{r+1}, i_{r+2}, \dots, i_n), T[k] = (r, r+1), E[k] = (i_r, i_{r+1}). \quad (14)$$

Введем обозначения:

$$A_x = Ai_1 \dots Ai_{r-1}, A_y = Ai_r, A_z = Ai_{r+1}, A_u = Ai_{r+2} \dots Ai_n. \quad (15)$$

Соединение отрезков на перестановке $Qx(k)$ в этих обозначениях:

$$A_{Qx(k)} = A_x A_y A_z A_u. \quad (16)$$

Тогда соединение отрезков на перестановке $Qx(k+1) = Qx(k) \mathrm{o} T[k]$:

$$A_{Qx(k+1)} = A_x A_z A_y A_u. \quad (17)$$

По замечанию 2 $i_r < i_{r+1}$. Согласно (9), $A_r \leq A_{r+1}$.Тогда, по утверждению (5), справедливо (13).

Таким образом, положение центра тяжести соединения отрезков, на перестановках «пузырек» не убывает.

Центр тяжести соединения отрезков на массиве «пузырек». Центр тяжести соединения отрезков определяется функцией $a(A)$ (см. формулу (6)). Предлагается вычислять эту функцию с помощью приращений на соседних перестановках «пузырек».

Теорема 2. Изменение (приращение) положения центра тяжести на транспозициях «пузырек».

Пусть имеется четыре отрезка:

$$A_x = (a_x, p_x, b_x), A_y = \left(a_y, p_y, b_y\right), A_z = (a_z, p_z, b_z), A_u = (a_u, p_u, b_u). \quad (18)$$

Введем обозначения для соединения отрезков:

$$A^1 = A_x A_y A_z A_u, A^2 = A_x A_z A_y A_u. \quad (19)$$

Определим размер (длину) отрезков:

$$d_x = a_x + b_x, d_y = a_y + b_y, d_z = a_z + b_z, d_u = a_u + b_u. \quad (20)$$

Определим общий вес отрезков:

$$p = p_x + p_y + p_z + p_{u.} \quad (21)$$

Отрезок A^2 получен из отрезка A^1 перестановкой (транспозицией) отрезков A_y и A_z: $A^2 = A^1 \mathrm{o}(y, z)$. Тогда $D(y, z) = a(A^2) - a(A^1)$ – изменение положения центра тяжести (приращение) на транспозиции (y, z). Оно вычисляется по формуле:

$$D(y, z) = a(A^2) - a(A^1) = (p_y d_z - p_z d_y)/p. \quad (22)$$

Доказательство.

Применяем формулу (6) вычисления центра тяжести при $n = 4$:

$$a(A^1) = [p_x a_x + p_y(a_y + d_x) + p_z(a_z + d_x + d_y) + p_u(a_u + d_x + d_y + d_z)]/p \quad (23)$$

$$a(A^2) = [p_x a_x + p_z(a_z + d_x) + p_y(a_y + d_x + d_z) + p_u(a_u + d_x + d_y + d_z)]/p \quad (24)$$

Вычисляя разность $a(A^2) - a(A^1)$, получаем требуемое – формулу (22).

Теорема 2 позволяет эффективно вычислять значения положения центра тяжести (функция a) на массиве перестановок «пузырек».

Алгоритм_2 вычисления центра тяжести. Предполагаем, что заданы n упорядоченных (см. формулу (9)) отрезков $A_1, \dots, A_n$. Построены массивы $T[m]$ и $E[m]$ по алгоритму_1, где $m = n(n-1)/2 + 1$.

Шаг 0. Определить массивы:

1) $D[m]$ – приращения функции a;

2) $S[m]$ – значения функции a.

По формуле (6) вычислить $a(A) = a(A_1A_2 \dots A_n)$ и $p(A)$.

Начальные значения массивов: $D[1] = 0$, $S[1] = a(A)$.

Шаг 1. Выполнить цикл:

```
FOR k=1 TO m-1
    i :=E[k+1, 1]
    j := E[k+1,2]
    D[k+1] :=[p(Ai)*( a(Aj)+b(Aj)) - p(Aj)*( a(Ai)+b(Ai))]/p(A)
    S[k+1] := S[k]+ D[k+1]
NEXT k
```

Шаг 2. Результат: массивы D и S. Конец.

Отметим, что временная сложность алгоритма_2: $O(n^2)$. На шаге «0» вычисление по формуле (6) имеет сложность $O(n^2)$ и, далее, на шаге «1» проход по всем элементам массива E: $O(n^2)$.

Пример 2. Для n = 5 созданы массивы T и E (табл. 1). В таблице 2 представлены данные отрезков A_1, A_2, A_3, A_4, A_5. Для иллюстрации отношения порядка на отрезках в таблице 2 показан их удельный вес γ.

Таблица 2 – Данные отрезков A_1, A_2, A_3, A_4, A_5

i	$a(A_i)$	$p(A_i)$	$b(A_i)$	γ_i
1	2.0000	7.0000	8.0000	0.7000
2	10.0000	21.0000	30.0000	0.5250
3	30.0000	12.0000	1.0000	0.3871
4	25.0000	10.0000	10.0000	0.2857
5	1.0000	3.0000	20.0000	0.1429

Результаты работы алгоритма_2 – массивы D и S представлены в таблице 3 одновременно с массивами T и E.

Таблица 3 – Массивы T, E, D, S

k	$T[k]$	$E[k]$	$D[k]$	$S[k]$
1	(1, 1)	(1, 1)	0.0000	52.9245
2	(1, 2)	(1, 2)	1.3208	54.2453
3	(2, 3)	(1, 3)	1.8302	56.0755
4	(3, 4)	(1, 4)	2.7358	58.8113
5	(4, 5)	(1, 5)	2.2075	61.0189
6	(1, 2)	(2, 3)	3.2264	64.2453
7	(2, 3)	(2, 4)	6.3208	70.5660
8	(3, 4)	(2, 5)	6.0566	76.6226
9	(1, 2)	(3, 4)	2.0755	78.6981
10	(2, 3)	(3, 5)	3.0000	81.6981

Оптимальная перестановка на массиве «пузырек». Задачу поиска оптимальной перестановки, соответствующей формуле (7), предлагается решать на массиве перестановок «пузырек» с помощью формирования транспозиций T, пар соответствующих переставляемых элементов E и вычисления центров тяжести S. При этом для фиксированного n (размерность задачи) можно один раз построить массивы T, E и, задавая разные варианты отрезков и целевого положения центра тяжести, решать множество задач поиска оптимальной перестановки.

Алгоритм_3 поиска оптимальной перестановки.

Шаг 0. Задать число n – количество размещаемых отрезков и начальную перестановку $Q = (1, 2,\ldots, n)$. Определить $m = n(n-1)/2+1$.

Шаг 1. Сформировать по алгоритму_1 массивы $T[m]$ и $E[m]$.

Шаг 2. Задать отрезки: $A_1 = (a_1, p_1, b_1)$, $A_2 = (a_2, p_2, b_2),\ldots, A_n = (a_n, p_n, b_n)$.

Для отрезков должно выполняться свойство, определяемое формулой (9): $A_1 \le A_2 \le \ldots \le A_n$.

Шаг 3. Сформировать по алгоритму 2 массивы $D[m]$, $S[m]$.

Шаг 4. Задать целевое положение центра тяжести: число c такое, что $S[1] \le c \le S[m]$. Отметим, что если $c < S[1]$ или $S[m] < c$, то решение очевидное: перестановки $Qx(1)$ или $Qx(m)$.

Шаг 5. Перебирая последовательно элементы массива S, найти k такое, что выполняется неравенство: $S[k] \le c \le S[k+1]$. Определить k_0: $k_0 = k$, если $(c - S[k]) \le (S[k+1] - c)$, иначе $k_0 = k + 1$.

Шаг 6. Результат – оптимальное решение задачи (7):

- перестановка $Qx(k_0) = Q \circ T[1] \circ T[2] \circ \ldots \circ T[k_0]$;
- положение центра тяжести на перестановке $Qx(k_0)$: $S[k_0]$;
- отклонение от цели $\delta = |S[k_0] - c|$.

Шаг 7. Конец.

Замечание 4. По теореме 1 $S[k] \le S[k+1]$ для всех $k = 1,2,\ldots,m - 1$. Тогда шаг 5 можно выполнить методом деления массива S пополам.

Замечание 5. Если после шага 6 необходимо скорректировать цель, то перейти на шаг 4.

Замечание 6. Если после шага 6 необходимо скорректировать отрезки, перейти на шаг 2.

Отметим, что в соответствии с замечаниями 5 и 6 алгоритмом 3 можно решать несколько различных задач одной размерности n. При этом массивы T и E на шаге 1 формируются один раз для всех этих задач.

Пример 3. Рассмотрим решение задачи (7) алгоритмом 3 на исходных данных примеров 1, 2. Результат шага 0: n = 5, Q = (1,2,3,4,5), m = 11. Результат шага 1 представлен в таблице 1. В таблице 2 результат шага 2 – заданные отрезки A_1, A_2, A_3, A_4, A_5. Результат шага 3 – в таблице 3. Выполним шаг 4: $c = 63$. Выполняем шаг 5. Результат легко определяется по таблице 3: $k = 5$, $S[5] \le c \le S[6]$. Подставляя числовые значения, определяем $k_0 = 6$. На шаге 6 получаем оптимальное решение:

– перестановка $Qx(6) = Q$o(1,1)o(1,2)o(2,3)o(3,4)o(4,5)o(1,2) = (3,2,4,5,1);
– положение центра тяжести на перестановке $Qx(6)$: $S[6] = 64.2453$;
– отклонение от цели $\delta = |S[6] - 63| = 1.2453$.

В работе [3] под точностью решения задачи (7) понимается отклонение от цели δ. Точность зависит от исходных данных. Если на перестановках «пузырек» $\delta = 0$, то найдено глобальное решение. Это возможно, в случае $c \in \{S[1], S[2], \ldots, S[m]\}$. Определим, при каком значении цели на перестановках «пузырек» будет достигаться максимальная величина δ. Пусть k_0 такое, что

$$D[k_0] = \max D[k], k = 1,2,\ldots,m. \tag{25}$$

Если $c = S[k_0] - D[k_0]/2$, то на исходных данных будет получено решение с максимальным отклонением от цели $\delta = D[k_0]/2$. В примере 3 по таблице 3 $k_0 = 7$ и $D[7] = 6.3208$ максимальный элемент в массиве D. Тогда, если целевое положение центра тяжести $c = S[7] - D[7]/2 = 67.4056$, то оптимальное решение на перестановке $Qx(7) = (3,4,2,5,1)$, центр тяжести $S[7] = 70.5660$, $\delta = 3.1604$.

Заключение. Данная работа является развитием результатов, изложенных в [3]. Приведено определение массива перестановок «пузырек». Предложен алгоритм формирования транспозиций, с помощью которых можно формировать перестановки этого массива. Доказаны следующие теоремы: теорема о неубывании положения центра тяжести соединения отрезков на перестановках «пузырек»; о вычислении приращения положения центра тяжести на соседних перестановках. На основе этих теорем построен алгоритм решения задачи размещения отрезков, нагруженных сосредоточенной силой. Временная сложность этого алгоритма $O(n^2)$, что на два порядка эффективнее алгоритма, предложенного в [3]. Приведены примеры работы алгоритма.

Библиографический список

1. Беллман Р. Прикладные задачи динамического программирования / Р. Беллман, С. Дрейфус. – М. : Наука, 1965. – 460 с.
2. Гэри М. Вычислительные машины и труднорешаемые задачи / М. Гэри, Д. Джонсон. – М. : Мир, 1982. – 416 с.
3. Дворянкин А. М. Задача размещения нагруженных отрезков/ А. М. Дворянкин, М. Б. Кульцова, И. Г. Жукова // Известия ВолгГТУ. Сер. Актуальные проблемы управления, вычислительной техники и информатики в технических системах. – Волгоград, 2017. – № 8 (203). – С. 29–34.
4. Залюбовский В. В. О представлении перестановками допустимых решений одномерной задачи упаковки в контейнеры / В. В. Залюбовский // Методы оптимизации : труды XIII Байкальской международной школы-семинара. – Иркутск : ИСЭМ СО АН РАН, 2005. – Т. 1. – С. 461–467.
5. Курейчик В. М. Алгоритмы одномерной упаковки элементов / В. М. Курейчик // Известия ЮФУ. Сер. Технические науки. – 2013. – № 7 (144). – С. 8–11.
6. Курош А. Г. Курс высшей алгебры / А. Г. Курош. – М. : Наука, 1968. – 431 с.
7. Мейер Б. Методы программирования : в 2 т. / Б. Мейер, К. Бодуэн ; пер. с фр. Ю. А. Первина ; под ред. А. П. Ершова. – М. : Мир, 1982. – Т. 2. – 368 с.
8. Kaluzny Bohdan L. Optimal aircraft load balancing / Bohdan L. Kaluzny, R. H. A. David Shaw // International Transactions in Operational Research. – November 2009. – Vol. 16, № 6. – P. 767–787 (21). DOI: https://doi.org/10.1111/j.1475-3995.2009.00723.x
9. Mongeau M., Be`s C., Optimization of aircraft container loading / M. Mongeau, C. Be`s // IEEE Transactionson Aerospace and Electronic Systems. – 2003. – Vol. 39, № 1. – P. 140–150.
10. Vancroonenburg W. Automatic air cargo selection and weight balancing: A mixed integer programming approach / W. Vancroonenburg, J. Verstichel, K. Tavernier, G. VandenBerghe // Transportation Research. Part E: Logistics and Transportation Review. – May 2014. – Vol. 65. – P. 70–83. DOI: https://doi.org/10.1016/j.tre.2013.12.013

References

1. Bellman R., Dreyfus S. *Prikladnye zadachi dinamicheskogo programmirovaniya* [Applied dynamic programming]. Moscow, Science Publ., 1965. 460 p.
2. Garey M., Johnson D. *Vychislitelnye mashiny i trudnoreshaemye zadachi* [Computers and inaccessible tasks]. Moscow, Mir Publ., 1982. 416 p.
3. Dvoryankin A. M., Kultsova M. B., Zhukova I. G. Zadacha razmeshheniya nagruzhennyh otrezkov [The problem of locating segments loaded by force]. *Izvestiya Volgogradskogo gosudarstvennogo tekhnicheskogo universiteta. Seriya «Aktualnye problem upravleniya, vychislitelnoy tekhniki i informatiki v tekhnicheskikh sistemakh»* [Proceedings of the Volgograd State Technical University. Series "Actual Problems of Control, Computer Technology and Computer Science in Technical Systems"], 2017, no. 8 (203), pp. 29–34.
4. Zalyubovskiy V. V. O predstavlenii perestanovkami dopustimyh resheniy odnomernoy zadachi upakovki v konteynery [On the representation by permutations of admissible solutions of the one-dimensional problem of packing into containers]. *Metody optimizatsii : trudy XIII Baykalskoy mezhdunarodnoy shkoly-seminara* [Optimization Methods : Proceedings of the XIII Baikal International School-Seminar]. Irkutsk, ISEM SB RAS, 2005, vol. 1, pp. 461–467.

5. Kureychik V. M. Algoritmy odnomernoj upakovkij elementov [Algoritms of 1-D elements packing], *Izvestiya JuFU. Seriya. Tehnicheskie nauki* [Izvestiya SFedU. Series. Engineering sciences], 2013, no.7 (144), pp. 8 – 11.

6. Kurosh A.G. Kurs vysshey algebry [The course of higher algebra]. Moscow, Science Publ., 1968. 431 p.

7. Meyer B., Baudoin C. *Metody programmirovaniya* [Programming methods]. In 2 vol. Moscow, Mir Publ., 1982. Vol. 2. 368 p.

8. KaluznyBohdan L., Shaw R. H. A. David. Optimal aircraft load balancing. *International Transactions in Operational Research*, November 2009, Vol. 16, no. 6, pp. 767–787 (21). DOI: https://doi.org/10.1111/ j.1475-3995.2009.00723.x

9. Mongeau M., Be`s C., Optimization of aircraft container loading. *IEEE Transactionson Aerospace and Electronic Systems*, 2003, vol. 39, no. 1, pp. 140–150.

10. Vancroonenburg W., Verstichel J., Tavernier K., VandenBerghe G. Automatic air cargo selection and weight balancing: A mixed integer programming approach. *Transportation Research. Part E: Logistics and Transportation Review*, May 2014, Vol. 65, pp. 70–83. DOI: https://doi.org/10.1016/j.tre.2013.12.013.

УДК 004.94

ВЛИЯНИЕ ФОРМЫ ПРЕДСТАВЛЕНИЯ ЭЛЕМЕНТОВ ТЕХНИЧЕСКОЙ СИСТЕМЫ В ПРОГРАММНОМ ОБЕСПЕЧЕНИИ НА СЛОЖНОСТЬ ЕГО АЛГОРИТМИЧЕСКОГО ОБЕСПЕЧЕНИЯ

Статья поступила в редакцию 12.01.2019, в окончательном варианте – 10.02.2019.

Алексеев Сергей Юрьевич, Тамбовский государственный технический университет, 392000, Российская Федерация, г. Тамбов, ул. Советская, 106,

кандидат технических наук, научный сотрудник, ORCID https://orcid.org/0000-0001-8748-5116, e-mail: sergej.alexejew@gmail.com

В работе исследовано влияние формы представления программных абстракций на сложность алгоритмов работы программного обеспечения. Выполнено сравнение двух форм представления абстракций с точки зрения их влияния на сложность алгоритмов работы программного обеспечения. Использованы формальные методы оценки двух вариантов структуры программного обеспечения, в основу которых положено использование метрик. Одна форма выполнена в соответствии с классическими представлениями, регламентируемыми объектно-ориентированными методами, вторая – на основе принципа, согласно которому абстракции должны быть выполнены максимально приближенными к терминам предметной области. Реализация этого принципа достигается за счет сочетания объектно- и проблемно-ориентированных методов конструирования программного обеспечения. В результате получается вариант структуры, который на первый взгляд не очевиден с точки зрения объектно-ориентированных техник, но очень близок к специфике предметной области и обладает преимуществом с точки зрения статических и динамических характеристик работы программного обеспечения. Программное обеспечение, составленное из абстракций, представляет собой программную систему, сценарий работы которой в общем случае складывается из сценариев работы каждой абстракции и сценария их взаимодействия. Чаще всего сценарий оперирует экземплярами сложных структур данных. Эти структуры данных и их экземпляры, реализуя различные формы представления данных, описывают элементы предметной области. От формы представления данных, формы и уровня абстрагирования при представлении в программной системе элементов прикладной области зависят статические и динамические характеристики алгоритмов ее работы, которые оперируют этими представлениями. Исследование выполнено на примере программного обеспечения для проектирования деревянных конструкций. Этот пример является частным, но он демонстрирует общие закономерности использования абстракций в сложных сценариях.

Ключевые слова: информационные технологии, повышение эффективности вычислений, программная абстракция технической системы, проектирование технической системы, метрики программного обеспечения, уровень сложности, надежность разработки

INFLUENCE OF THE FORM OF REPRESENTATION OF THE ELEMENTS OF THE TECHNICAL SYSTEM IN THE SOFTWARE ON THE DIFFICULTY OF ITS ALGORITHMIC ENVIRONMENT

The article was received by editorial board on 12.01.2019, in the final version – 10.02.2019.

Alekseev Sergey Yu., Tambov State Technical University, 106 Sovetskaya St., Tambov, 392000, Russian Federation,

Cand. Sci. (Engineering), Researcher, ORCID https://orcid.org/0000-0001-8748-5116, e-mail: sergej.alexejew@gmail.com

The paper studies the influence of the form of representation of software abstractions on the complexity of software algorithms. A comparison of two forms of representation of abstractions from the point of view of their influence on the complexity of software algorithms has been performed. One form is made in accordance with the classical ideas regu-