Скористаємось відомою формулою [3]

$$\int_0^{\infty} \frac{J_0(rR)\,R}{R^2+z^2}dR = K_0(rz), \tag{3}$$

де $K_0(rz)$ – модифікована функція Бесселя другого роду (функція Макдональда) порядку нуль. Далі помножимо ліву та праву частину інтегралу (3) на r та візьмемо інтеграл від 0 до r. Знову використовуючи співвідношення [3] отримаємо

$$\int_0^r rJ_0(rR)dr = \frac{r}{R}J_1(rR), \qquad \int_0^r rK_0(rz)dr = -\frac{r}{z}K_1(rz)+\frac{1}{z^2},$$

$$\int_0^{\infty} \frac{J_1(rR)}{R^2+z^2}dR = -\frac{1}{z}\cdot K_1(rz)+\frac{1}{r\cdot z^2}. \tag{4}$$

Таким чином, коректне значення інтегралу (2) отримано в (4). Обчислення у системі Maple 17 дає результат

```
[> r:=0.5: z:=0.256:
[> -1/z*BesselK(1,r*z) + 1/r/z**2;
                    0.6696673629
```

Отримане значення узгоджується із таблицею, що наведена вище.

Повторюючи n-раз зазначену процедуру, отримаємо формулу для інтегрування (1)

$$\int_0^{\infty} \frac{R^{\nu-n+1}J_{\nu+n}(rR)}{R^2+z^2}dR = (-1)^n z^{\nu-n}K_{\nu+n}(rz)-\frac{1}{2}\sum_{k=1}^{n}(-1)^k \frac{z^{-2k}\Gamma(\nu+k)}{\Gamma(n-k+1)}\cdot\left(\frac{r}{2}\right)^{n-\nu-2k}.$$

Вперше алгоритм аналітичного обчислення інтегралів типу (1) був представлений у [4].

Робота виконана у рамках держбюджетної теми «Розробка методів дослідження міцності та стійкості тонкостінних оболонок та пружних твердих тіл з рідиною при дії різного виду динамічних навантажень» (№ держреєстрації 0119U100042)

Література

1. Шевченко В. П. Методы фундаментальных решений в теории ортотропных оболочек. *Концентрация напряжений.* К. : А.С.К., 1998 С. 205–207. (Механика композитов: В 12 т.; т. 7).

2. Маричев О. И. Метод вычисления интегралов от специальных функций (теория и таблицы формул). Минск : Наука и техника, 1978. 310 с.

3. Бейтмен Г., Эрдейи А. Высшие трансцендентные функции. Том 2. М. : Наука, 1974. 296 с.

4. Величко П. М., Шевченко В. П. О действии сосредоточенных сил и моментов на оболочку положительной кривизны. *Изв. АН СССР. механика твердого тела.* 1969. № 2. С. 147–151.

УДК 519.688:004.02:004.9

КОМП'ЮТЕРНО-МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ МОЖЛИВОСТЕЙ КОРЕКЦІЇ ВИЗНАЧЕННЯ ПЕРЕМОЖЦЯ ГОЛОСУВАННЯ МЕТОДОМ БОРДА

О. С. Вєтров, К. М. Довбня, Д. О. Ливицька

Одним із найуживаніших на практиці методів голосування є метод Борда [1]. На сьогодні метод Борда – це не один, а ціла група методів, що мають єдину ідейну основу, і розрізняються у певних деталях реалізації.

Метод Борда – це система голосування з єдиним переможцем, у якій кожен виборець ранжує список кандидатів в порядку переваги, тобто метод Борда відноситься класу систем преференціального голосування (як і, наприклад, метод Кондорсе). Після того, як виборець впорядкував кандидатів в порядку своїх вподобань, він присвоює p_1 балів своєму найкращому кандидату, p_2 балів наступному у рейтингу і так далі до останнього кандидата із p_k балами (вважаємо $p_1 > p_2 > ... > p_{k-1} > p_k$). Отримані бали кандидата підсумовуються за усіма виборцями. Кандидат, який набрав найбільшу суму балів, перемагає.

Нехай $N = \{1, n\}$ – множина виборців, $C = \{c_i\}_{i=1}^{k}$ – множина кандидатів. Система індивідуальних переваг усіх виборців називається профілем.

Дані для прикладу розглянемо наступний профіль

Таблиця 1

Кількість голосів	**8**	**5**	**2**	**4**	**2**	**9**
Впорядкування кандидатів	c_1	c_1	c_2	c_2	c_3	c_3
	c_2	c_3	c_1	c_3	c_1	c_2
	c_3	c_2	c_3	c_1	c_2	c_1

Позначимо $m(c_i)$ загальну суму балів *i*-го кандидата, v_i – кількість голосів у профілі за *i*-ту конфігурацію (впорядкування) кандидатів. Тоді для профілю, заданого таблицею 1, можна записати

$$P = \begin{pmatrix} p_1 & p_1 & p_2 & p_3 & p_2 & p_3 \\ p_2 & p_3 & p_1 & p_1 & p_3 & p_2 \\ p_3 & p_2 & p_3 & p_2 & p_1 & p_1 \end{pmatrix}, \quad v = (v_1, v_2, v_3, v_4, v_5, v_6),$$

$$M = \begin{pmatrix} m(c_1) \\ m(c_2) \\ m(c_3) \end{pmatrix} = P \cdot v^T .$$

Задача пошуку кандидата-переможця зводиться до знаходження $\max(M)$.

Коефіцієнти p_i можна обирати різними способами. Найбільш поширеними є два [2]:

- класичний, коли $p_1 = k,\ p_2 = k-1,\ ...\ , p_k = 1$.
- починаючи з нуля, тобто $p_1 = k-1,\ p_2 = k-2,\ ...\ , p_k = 0$.

Рідше згадується так звана Dowdall System [2], система, в якій $p_k = 1/k$. При цьому згаданий підхід реально використовується під час голосування у державах Словенія та Науру.

Припустимо ситуацію, що експерти вирішили запропонувати свою систему розподілу балів за зайняті у профілі місця кандидатами з єдиним обмеженням на значення p_k: обов'язково має виконуватись система нерівностей $p_1 > p_2 > ... > p_{k-1} > p_k$.

В таблиці 2 наведені різні приклади значення коефіцієнтів p_1, p_2 та p_3, що дозволяють на профілі з таблиці 1 «обирати» переможця.

Таблиця 2

#	p_1	p_2	p_3	$m(c_1)$	$m(c_2)$	$m(c_3)$
1.	5	2	1	**<u>86</u>**	71	83
2.	5	3	1	90	88	**<u>92</u>**
3.	5	4	1	94	**<u>105</u>**	101
4.	10	2	1	**<u>151</u>**	101	138
5.	10	5	1	163	152	**<u>165</u>**
6.	10	7	1	171	**<u>186</u>**	183
7.	3	2	1	60	59	**<u>61</u>**
8.	2	1	0	30	29	**<u>31</u>**
9.	1	1/2	1/3	**<u>19,(3)</u>**	16,8(3)	18,8(3)

Можна зробити висновок, що для запобігання фальсифікації голосування коефіцієнти p_k мають бути заданими до самого голосування.

В подальшому дослідженні планується аналітично отримати співвідношення коефіцієнтів p_k, що для даного вектору v дозволяли би моделювати переможця. Починаючи з $k = 3$, в подальшому отримані співвідношення будуть поширені на випадок довільного натурального k. Також засобами комп'ютерно-математичного моделювання буде визначена залежність окремими конфігураціями упорядкування кандидатів та відповідними компонентами вектору v, з метою формування рекомендацій до вибору коефіцієнтів p_k.

Література

1. Волошин О. Ф., Мащенко С. О. Моделі та методи прийняття рішень. К.: Видавничо-поліграфічний центр «Київський університет», 2010. 336 с.
2. Emerson P. The original Borda count and partial voting. *Social Choice and Welfare.* 2013. Volume 40, issue 2. Pages 353–358.
3. Fraenkel J., Grofman B. The Borda Count and its real-world alternatives: Comparing scoring rules in Nauru and Slovenia. *Australian Journal of Political Science.* 2014. Volume 49, issue 2. Pages 186–205.

УДК 519.683:004.02:004.9

ПРОГРАМНА РЕАЛІЗАЦІЯ ЕФЕКТИВНОГО АЛГОРИТМУ ГРИ У «БАЛДУ»

О. В. Мазурук

«Балда» – це лінгвістична гра, в якій необхідно складати слова за допомогою літер, що додаються певним чином на квадратне ігрове поле. Найчастіше правила є такими, що слова складаються за допомогою переходів від букви до букви під прямим кутом. В мережі можна знайти декілька підходів до програмної реалізації гри «Балда» [1, 2]. Очевидно, що великою мірою ефективність програми залежить від словника (його обсягу, схеми організації – спосіб зберігання словника у пам'яті комп'ютера тощо).

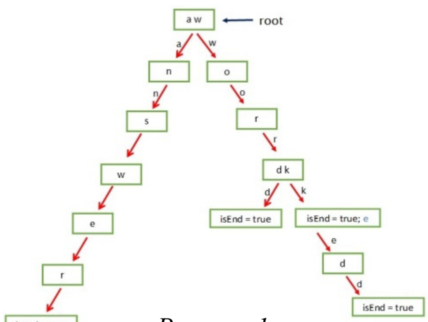

Рисунок 1

Ефективність роботи програми буде залежати від оптимально обраної структури даних для опису словника. При завантаженні кожне слово записується в структуру у вигляді дерева (рис. 1), де кожен вузол цього дерева означає конкретну букву і посилається не більш ніж на 26 піддерев (обраний латинський алфавіт). Нижче описаний код class Node – вузол дерева (мова програмування Java).

```
class Node {
    Node[] subsidiaryNodes;
    boolean isEnd;
    public Node() {
        this.subsidiaryNodes = new Node[26];
    }
}
```

тут subsidiaryNodes – масив для зберігання дочірніх вузлів, isEnd – булева змінна, яка перевіряє, чи є поточний вузол – листом.