УДК 519.6

Кисіль Т.М.
Хмельницький національний університет

Бедратюк Г.І.
Хмельницький національний університет

ВИКОРИСТАННЯ ІНТЕРАКТИВНОГО СЕРЕДОВИЩА MATLAB ДЛЯ РОЗВ'ЯЗАННЯ ЗАДАЧ ОПТИМІЗАЦІЇ

У статті подано опис основних функцій середовища Matlab та інструметарію Optimization Toolbox для розв'язання задач оптимізації. Розглянуто способи розв'язання деяких типових задач лінійної та нелінійної оптимізації в Matlab. Зокрема, розглянуто команди для таких розділів: задачі лінійного програмування, транспортні задачі, задачі нелінійної оптимізації.

Ключові слова: *цільова функція, мінімум, максимум, оптимізація, Matlab, Optimization Toolbox.*

Постановка проблеми. Сучасний етап розвитку суспільства зумовлює високі вимоги до оцінювання ефективності функціонування систем різної природи. До завдань оптимізаційного типу належать завдання, в яких потрібно знайти найкраще або оптимальне рішення при заданих умовах. Особливістю завдань оптимізаційного типу є багатоваріантність їх рішень, що зумовлена різноманітністю факторів, які можуть бути розглянуті. Важливість задач оптимізаційного типу є безумовна. Розв'язання таких задач особливо в умовах невизначеності здебільшого є складним питанням навіть для математиків. У зв'язку з цим дедалі частіше доводиться використовувати комп'ютерні технології, спеціалізовані пакети команд, додатковий інструментарій [1–3].

MATLAB – це платформа для програмування, розроблена спеціально для інженерів і вчених. MATLAB поєднує в собі середовище, налаштоване для багатократного аналізу та дизайну процесів, і мову програмування, яка безпосередньо використовує матриці й математичні масиви [1].

Практично для кожного розділу математики в MATLAB розроблено окремий спеціалізований пакет команд, зокрема ***Optimization Toolbox. Optimization Toolbox*** надає широкий набір алгоритмів для вирішення стандартних задач оптимізації та задач оптимізації великої розмірності. Ці алгоритми вирішують завдання з обмеженнями й без обмежень, з неперервним і дискретним аргументом. ***Optimization Toolbox*** має функції для лінійного програмування, квадратичного програмування, бінарного цілочисельного програмування, нелінійної оптимізації, нелінійного методу найменших квадратів, систем нелінійних рівнянь і багатокритеріальної оптимізації [7].

Аналіз останніх досліджень і публікацій. Останнім часом спостерігається активне проникнення спеціалізованих математичних пакетів в освітній процес, оскільки це дає можливість формування принципово нових технологій навчання [2], уникати рутини. Проте натепер ці технології, незважаючи на свою ефективність і наочність, у силу різних причин ще недостатньо поширені в навчальному процесі.

Необхідні знання та початкові навики роботи в середовищі MATLAB детально розглянуто в роботі В.П. Дьяконова [1].

Стаття є продовженням статей [4–5], спрямованих на інформатизацію й комп'ютеризацію освіти. Матеріали статті можуть бути використані студентами та викладачами ВНЗ для навчальних цілей, зокрема розв'язання типових задач, які зустрічаються в процесі вивчення дисциплін «Методи оптимізації», «Математичне моделювання», «Комп'ютерне моделювання» тощо.

Постановка завдання. Метою статті є розгляд основних команд і спеціалізованого набору інструментів **Optimization Toolbox** середовища Matlab, який може бути використаний для розв'язання типових задач лінійної та нелінійної оптимізації.

Виклад основного матеріалу дослідження.

1. Задачі лінійного програмування.

Нагадаємо, що основною задачею лінійного програмування [6] є знаходження мінімуму цільової функції

$$f = c_1x_1 + c_2x_2 + \cdots + c_nx_n,$$

за таких лінійних обмежень на змінні $x_1, x_2, \dots x_n$:

$$\begin{cases} a_{1,1}x_1 + a_{1,2}x_2 + \cdots a_{1,n}x_n \le b_1, \\ a_{2,1}x_1 + a_{2,2}x_2 + \cdots a_{2,n}x_n \le b_2, \\ \vdots \\ a_{m,1}x_1 + a_{m,2}x_2 + \cdots a_{m,n}x_n \le b_m. \end{cases}$$

У матричному записі задача лінійного програмування записується у вигляді

$$A \cdot X \le B, f = (C, X) \to \min,$$

для деякої матриці A та векторів X, B, C. Якщо функція f не є лінійною або обмеження на змінні не є лінійними, то задача знаходження її мінімуму називається задачею нелінійної оптимізації.

Розглядати задачі оптимізації зручно в матричному вигляді, тим більше що «серцем» MATLAB є матриці.

Для ініціалізації основних матриць і розвязання задачі необхідно:

1) згрупувати всі невідомі в один вектор *(x)*;
2) відокремити межі (верхні й нижні) значень змінних; записати вектори верхніх *(ub)* і нижніх меж *(lb)*;
3) записати матрицю *(A)* і вектор *(b)* для обмежень – нерівностей;
4) записати матрицю (A_{eq}) і вектор (b_{eq}) для обмежень – рівностей;
5) записати цільову функцію;
6) викликати потрібний солвер та отримати розвязок.

Залежно від типу оптимізаційної задачі й методу отримання розвязку, викликається та чи інша команда, зокрема для задачі лінійного програмування викликається солвер linprog, для випадку, коли цільова функція є квадратична, викликаємо солвер quadprog [1; 7].

Одним із варіантів синтаксису названих команд є такий.

[X, FVAL] = linprog (f,A,b,Aeq,beq,LB,UB) вирішує задачу лінійного програмування з обмеженнями типу рівності та нерівності за умови, що шукані змінні є обмежені зверху і знизу. Повертає вектор розв'язок, що буде записаний у X, і значення цільової функції FVAL.

[X, FVAL] = quadprog (f,A,b,Aeq,beq,LB,UB) вирішує задачу квадратичного програмування з обмеженнями типу рівності та нерівності за умови, що шукані змінні є обмежені зверху і знизу. Повертає вектор розв'язок, що буде записаний у X, і значення цільової функції FVAL.

Алгоритм розв'язання типової задачі лінійного програмування розглянемо на абстрактному прикладі, не привязуючись до прикладного змісту змінних і параметрів.

Приклад 1. Знайти мінімум функції $f = 3x_1 + 5x_2 + 12x_3,$ при таких обмеженнях:

$$\begin{cases} 2x_1 + x_2 + 4x_3 \ge 20, \\ 4x_1 + 5x_2 \ge 40, \\ x_1, x_2, x_3 \ge 0. \end{cases}$$

Маємо вектор (x_1, x_2, x_3). Усі змінні додатні, тому вектор нижніх меж:

lb=zeros(3,1).

Зверху на змінні в нашому випадку обмежень не накладено, тому вектор верхніх меж:

lb=Inf(3,1).

Задамо тепер матрицю й вектор для обмежень типу нерівностей:

A=[-2 -1 -4; -4 -5 0];

b=[-20; -40].

Обмежень типу рівностей за умовою задачі не передбачено, але в разі звернення до потрібної функції MATLAB вона має бути означена, тому обираємо порожні матрицю і вектор для обмежень типу рівностей:

Aeq=[];

beq=[].

Тепер задамо цільову функцію:

f=[3 5 12].

Усі початкові ініціалізації зручно робити в скриптовому файлі, щоб за потреби була можливість їх швидко модифікувати.

```
MuOptim.m
%нижні і верхні межі
lb=zeros(3,1);
ub=Inf(3,1);
%матриця і вектор для обмежень-нерівностей
A=[-2 -1 -4; -4 -5 0];
b=[-20; -40];
%матриця і вектор для обмежень-рівнянь
Aeq=[];
beq=[];
%цільова функція
f=[3 5 12];
```

Рис. 1. Скрипт для визначення вхідних параметрів

Викличемо потрібний солвер (із командного вікна після виконання скрипта) й отримаємо розв'язок.

```
Command Window
>> [x fval]=linprog(f,A,b,Aeq,beq,lb,ub)

Optimal solution found.

x =
    10
     0
     0
fval =
    30
fx >>
```

Рис. 2. Результати роботи солвера

Отже, $x_{опт} = (10, 0, 0); f_{опт} = 30$.

Для можливості використання набору інструментів **Optimization Toolbox** середовища Matlab

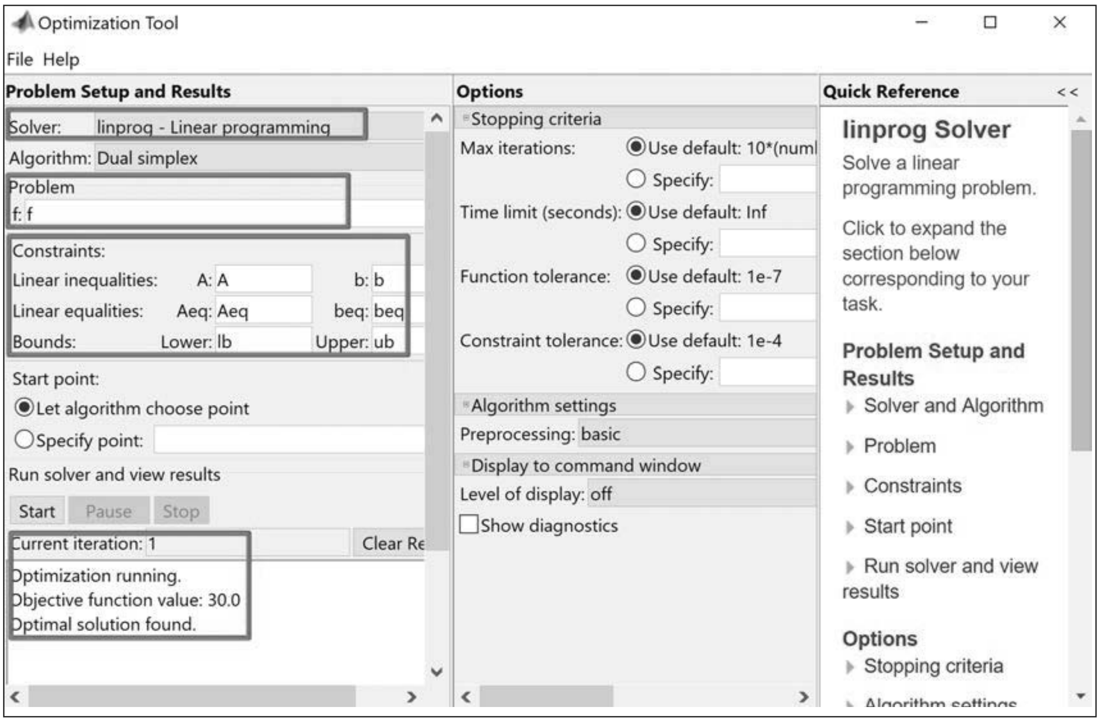

Рис. 3. Користувацький інтерфейс Optimization Toolbox

потрібно в командному вікні MATLAB набрати командний рядок такого вигляду:

>> optimtool

або натиснути відповідну піктограму в меню APPS.

optimtool запускає графічний користувацький інтерфейс для розв'язання оптимізаційних задач у MATLAB. optimtool може бути використаний для редагування параметрів за замовчуванням, вибору та запуску програми-солвера.

Приклад 2. Під час розв'язання попередньої задачі використати набір інструментів **Optimization Toolbox** середовища MATLAB.

У вікні, що відкрилось, обираємо потрібний солвер, задаємо цільову функцію й обмеження. (Зауважимо, що необхідні параметри після виконання скриптового файлу перенесені в робочий простір, тому можемо їх використовувати.) Запускаємо пошук розв'язку й отримуємо результат.

Зручність і зрозумілість інтерфейсу інструментарію Optimization Toolbox очевидна.

2. Транспортна задача.

Нехай у пунктах $A_1, A_2, ..., A_m$ виробляється деякий однорідний продукт, причому обсяг виробництва цього продукту в пункті A_i a_i одиниць. Зроблений у пунктах виробництва продукт повинен бути доставлений до пунктів споживання $B_1, B_2, ..., B_n$, причому обсяг споживання в пункті B_j становить b_j одиниць продукту. Уважається, що транспортування готової продукції можливе з будь-якого пункту виробництва в будь-який пункт споживання, і транспортні витрати, що припадають на перевезення одиниці продукту з пункту A_i в пункт B_j становлять c_{ij} грошових одиниць. Задача полягає в організації такого плану перевезень, за якого сумарні транспортні витрати були б мінімальними [6].

Формально задача ставиться так. Нехай x_{ij} – кількість продукту, що перевозиться з пункту A_i в пункт B_j. Потрібно визначити сукупність із mn величин x_{ij}, які відповідають умовам:

1. $\sum_{j=1}^{n} x_{ij} = a_i, \quad \forall i$;
2. $\sum_{j=1}^{m} x_{ij} = b_j, \quad \forall j$;
3. $x_{ij} \geq 0, \quad \forall i \forall j$

і для яких лінійна форма $z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$ набуває найменшого значення.

Приклад 3. Розв'язати транспортну задачу:

	B_1	B_2	B_3	B_4	Кількість
A_1	11	25	13	43	160
A_2	14	31	22,5	10	180
A_3	20	22	28	21	300
Кількість	140	60	180	260	640

Позначимо через $x_{i,j}$ кількість продукції, яку нам потрібно перевезти з пункту A_i в пункт B_j. Тоді нам необхідно мінімізувати таку цільову функцію

```
Command Window
>> my_trans
Введіть матрицю тарифних коефіцієнтів разом із векторами постачання і споживання ...
[11 25 13 43 160; 14 31 22.5 10 180; 20 22 28 21 300; 140 60 180 260 640]
Матриця перевезень:
   140    20     0     0
     0    40   140     0
     0     0    40   260

Значення цільової функції 13010
```

Рис. 5. Результат розв'язання поставленої задачі

$$f = 11x_{11} + 25x_{12} + 13x_{13} + 43x_{14} + 14x_{21} + 31x_{22} +$$
$$+ 22.5x_{23} + 10x_{24} + 20x_{31} + 22x_{32} + 28x_{33} + 26x_{34}$$

при таких обмеженнях (задача закритого типу)

$$\begin{cases} x_{11} + x_{21} + x_{31} = 140, \\ x_{12} + x_{22} + x_{32} = 60, \\ x_{13} + x_{23} + x_{33} = 180, \\ x_{14} + x_{24} + x_{34} = 260, \\ x_{11} + x_{12} + x_{13} + x_{14} = 160, \\ x_{21} + x_{22} + x_{23} + x_{24} = 180, \\ x_{31} + x_{32} + x_{33} + x_{34} = 300 \end{cases}.$$

Нагадаємо, що в MATLAB дуже зручно працювати з матрицями, тому, не применшуючи важливості вже розглянутих убудованих функцій і згаданого Optimization Toolbox, створимо свою функцію, яка розв'яже нашу транспортну задачу методом північно-західного кута [6].

```
function [outputArg1,outputArg2] = my_trans(inputArg1,inputArg2);
C=input ('Введіть матрицю тарифних коефіцієнтів разом із векторами постачання і споживання ');
[m n]=size(C);
x=zeros(m,n);%шукана матриця
sumc=0;
sumr=0;
for i=1:m-1
sumc=sumc+C(i,n);
end
for j=1:n-1
sumr=sumr+C(m,j);
end
if(sumc == sumr)
for i=1:m
for j=1:n
a=min(C(i,n),C(m,j));
x(i,j)=a;
C(i,n)=C(i,n)-a;
C(m,j)=C(m,j)-a;
end
end
else disp('не виконується умова балансу');
end
disp(x(1:(m-1),1:(n-1)));
xre=0;
 or i=1:m-1
for j=1:n-1
xre=xre+(C(i,j).*x(i,j));
end
end
disp(['Значення цільової функції',num2str(xre)]);
end
```

Рис. 4. Реалізація методу північно-західного кута

Виконаємо цю функцію і протестуємо на цій задачі.

Отже, $x_{11} = 140,\ x_{12} = 20,\ x_{22} = 40,\ x_{23} =$ $= 140,\ x_{33} = 40,\ x_{34} = 260,\quad \textit{інші}\ x_{ij} = 0,\quad f_{опт} = 13010$.

3. Нелінійні задачі.

Нелінійні задачі можуть бути розв'язані за допомогою команд пакету **Optimization Toolbox.** Команда **fminunc** дає змогу знайти глобальний (локальний) мінімум цільової функції багатьох змінних без обмежень і з обмеженнями. Зауважимо, що цільова функція та обмеження можуть бути нелінійними.

Приклад 4. Знайдемо мінімум функції $f = 5\sqrt{x} + \sqrt[3]{y}$

```
>> x = fminunc(@(x) 5*x(1)^(1/2) + x(2)^(1/3),[64;64])
x =
0.0000
59,7333.
```

Команда fmincon знаходить екстремум нелінійної функції багатьох змінних із заданими обмеженнями.

Приклад 5. Знайдемо умовний мінімум функції $f = 100(y - x^2)^2 + (1 - x)^2$ за умови $x + 2y \le 1$, обрати початкові значення змінних (-1; 2).

```
>> fun = @(x)100*(x(2)-x(1)^2)^2 + (1-x(1))^2;
>> x0 = [-1,2];
A = [1,2];
b = 1;
x = fmincon(fun,x0,A,b) x =
0.5022    0.2489.
```

Висновки. У статті подано опис деяких основних команд інструметарію **Optimization Tools** системи MATLAB. Розглянуто способи розв'язання деяких типових задач лінійної та нелінійної оптимізації. Використовуючи розглянуті команди пакету MATLAB, можна ілюструвати розв'язання задач на практичних заняттях задля запобігання рутинним діям або для порівняння результатів обчислень під час вивчення таких курсів, як методи оптимізації та комп'ютерного моделювання тощо.

Список літератури:

1. Дьяконов В.П. MATLAB. Полный самоучитель. Москва: ДМК Пресс, 2012. 768 с.: ил.
2. Крупський Я.В. Активізація навчально-пізнавальної діяльності студентів при вивченні диференційного числення (за допомогою Maple-технологій). Сучасні інформаційні технології та інноваційні методики навчання в підготовці фахівців: методологія, теорія, досвід, проблеми. 2010. Вип. 26. С. 339–344.
3. Алгоритмы. Построение и анализ / Т. Кормен, Ч. Лейзерсон, Р. Ривест, К. Штайн. Москва: Вильямс, 2013. 1324 с.
4. Бедратюк Г.І. Використання Maple при вивченні дисципліни «Методи синтезу та оптимізації». Збірник наукових праць ФПМКТ. Хмельницький: ХНУ, 2010. № 1 (3). С. 137–141.
5. Бедратюк Л.П., Бедратюк Г.І. Системи комп'ютерної алгебри в теорії графів. Східноєвропейський журнал передових технологій. 2012. Т. 6. № 4 (60). С. 43–46.
6. Дослідження операцій: конспект лекцій / уклад.: О.І. Лисенко, І.В. Алєксєєва. Київ: НТУУ «КПИ», 2016. 196 с.
7. Optimization Toolbox User's Guide @ COPYRIGHT 1990–2003 by The MathWorks, Inc. URL: http://www.ing.unitn.it/ ~afanasye/ 10_UniTN_reports/ 20_Presentations/20_Lettures/Robotica_e_Sensor_Fusion_lezioni_2012/Lezione%203%20-%20Optimisation/materials/optim_tb.pdf (дата звернення: 18.09.2018).

ИСПОЛЬЗОВАНИЕ ИНТЕРАКТИВНОЙ СРЕДЫ MATLAB ДЛЯ РЕШЕНИЯ ЗАДАЧ ОПТИМИЗАЦИИ

В статье дано описание некоторых функций среды Matlab и инструмента Optimization Toolbox для решения задач оптимизации. Рассмотрены способы решения некоторых типовых задач линейной и нелинейной оптимизации в Matlab. В частности, рассмотрены команды для следующих разделов: задачи линейного программирования, транспортные задачи, задачи нелинейной оптимизации.

***Ключевые слова:** целевая функция, минимум, максимум, оптимизация, Matlab, Optimization Toolbox.*

USING THE MATLAB INTERACTIVE SOURCE FOR SOLVING OPTIMIZATION PROBLEMS

A description of Matlab's core functions and the Optimization Toolbox toolkit for solving optimization problems is given. Methods of solving some typical linear and nonlinear optimization problems in Matlab are considered. In particular, the commands for the following sections are considered: linear programming problem, transportation problem, problems of nonlinear optimization.

***Key words:** objective function, minimum, maximum, optimization, Matlab, Optimization Toolbox.*