

9.2 Моделювання інформаційно-комунікаційних систем методами математичного програмування

При розробці інформаційно-комп'ютерних систем управління технічними процесами, при проектуванні роботи виробництва вирішується питання вибору найкращого варіанту роботи підприємства або певних процесів з точки зору мінімізації або максимізації значень певних показників. Моделювання виконання технічних процесів у системі починається з постановки задачі, у ході якої обирається цільова функція, як аналітичний запис функціональної залежності від певних показників. При математичній постановці задачі вказується яке значення функції цілі вважаються оптимальними. Також записуються певні умови та обмеження, які накладаються на аргументи функції, тобто на показники як параметри.

Якщо проблема вибору може бути вирішена багатьма шляхами, виникає необхідність порівняння варіантів, які обираються або проектуються, або рішень, що приймаються. Для цього необхідно мати можливість оцінити, які варіанти кращі, які гірші. Для порівняння варіантів, які оцінюються, користуються поняттям ефективності, що є узагальнюючою характеристикою об'єктів як систем, спрямованих на вирішення певних функцій.

Під ефективністю часто розуміють головну, визначальну характеристику або сполучення найбільш важливих техніко-економічних характеристик. У кібернетиці ефективність якісно повинна виражати пристосованість системи до функціонування за певним алгоритмом і кількісно повинна визначатися функцією, що виражає співвідношення між метою, що досягається, та витратами.

Якщо розглядати систему як об'єкт з точки зору системного аналізу, використовуючи методи системного аналізу, то увесь комплекс параметрів, що впливають на ефективність роботи системи можна поділити на показники, при збільшенні значень яких, збільшується значення функції ефективності, поліпшуються позитивні властивості технічної системи, та показники, збільшення значень яких, погіршує властивості системи, посилюються шкідливі

характеристики якості системи.

Показники першого класу назвемо позитивними, другого – негативними. До позитивних показників відносяться: у випадку технічних систем зв'язку - вірогідність і швидкість передачі інформації, надійність системи тощо, до негативних - апаратурна складність системи, її вартість, займана смуга частот тощо; у випадку промислових підприємств, що виготовляють продукцію – до позитивних показників відносяться: товарна продукція, сума прибутку; до негативних – кількість працівників, матеріальні витрати, основні виробничі фонди тощо.

Коли показник один, то вибрати просто: найкращий варіант той, для якого показник, значення якого бажано збільшити (позитивний), має більше числове значення, а показник, значення якого збільшувати небажано, (негативний) – має найменше значення. У загальному випадку позитивні і негативні показники є суперечливими і взаємозалежними: збільшення значення кількісної міри позитивних показників може привести до збільшення значення негативних показників. Протиріччя можуть мати місце як між позитивними і негативними показниками, так і між позитивними і позитивним чи між негативними і негативними.

Таким чином, ставиться задача формування критерію ефективності, що буде виражати узагальнено, в сукупності, характеристики об'єкту.

Введення понять позитивних і негативних показників дає можливість виразити весь комплекс показників системи одним відносно простим математичним співвідношенням з урахуванням коефіцієнтів важливості кожного показника. Дійсно, при такому визначенні показників можна вважати, що головна задача синтезу оптимальної системи полягає в досягненні найбільшої суми значень позитивних показників і найменшої суми значень негативних показників з урахуванням важливості кожного показника. Далі природно припустити, що з усіх варіантів, кожний з яких має як позитивні, так і негативні показники з різними значеннями, найкращим для спостерігача буде той варіант, що задовольняє заданим вимогам щонайкраще.

Таким чином, критерій має бути таким, щоб при загальному збільшенні показників, що виражають функцію цілі системи, значення критерію зростало, а при зростанні сукупних витрат – зменшувалося. При наведеному вище визначенні двох видів показників системи ціль, що досягається, (якість системи) характеризує сума значень позитивних показників, а витрати - сума значень негативних показників. Усі ці показники в усіх без винятку випадках можуть бути визначені кількісно.

Отже, весь комплекс показників системи необхідно задати одним сукупним параметром F_0 – функцією ефективності, що виражається через значення позитивних X_i і значення негативних Y_j показників з урахуванням важливості кожного показника. При цьому сукупний параметр F_0 повинний збільшуватися зі збільшенням суми значень позитивних показників $\sum_{i=1}^n X_i$ і зменшуватися зі збільшенням суми значень негативних показників $\sum_{j=1}^m Y_j$, де m і n – кількість позитивних і негативних показників системи відповідно ($i=1,2,...,n$; $j=1,2,...,m$).

Оскільки кожен показник X_i і Y_j має різну важливість в системі заданих вимог (зовнішніх умов), то показники X_i і Y_j повинні бути перемножені на певні вагові коефіцієнти важливості a_i і b_j відповідно. Вагові коефіцієнти a_i і b_j виражають кількісну міру важливості позитивних і негативних показників відповідно.

Таким чином, сукупний параметр F_0 можна представити наступною функцією:

$$F_0 = f\left(\sum_{i=1}^n a_i \cdot X_i, \sum_{j=1}^m b_j \cdot Y_j\right) \quad (1)$$

У цьому співвідношенні сума $\sum_{i=1}^n a_i \cdot X_i$ являє собою кількісну величину

якості системи, а сума $\sum_{j=1}^m b_j \cdot Y_j$ – сукупні витрати.

Це ж співвідношення виражає конкретне число й охоплює всі показники системи з урахуванням кількісної міри значень і важливості показників системи.

Критерій ефективності також може розглядатися як лінійна комбінація кількісних показників:

$$F = \sum_i a_i \cdot X_i - \sum_j b_j \cdot Y_j \quad (2)$$

де a , b – вагові коефіцієнти показників.

В принципі, сформувати функцію критерію ефективності можливо також в іншій формі, наприклад, просто, об'єднавши позитивні та негативні показники разом:

$$F = \sum_i a_i \cdot X_i \quad (3)$$

У якості методів прийняття математично-обґрунтованих техніко-економічних рішень використовуються методи лінійного програмування, вирішуються виробничі задачі симплекс-методом, перевіряються результати розрахунку на ЕОМ. Задачі лінійного програмування відносяться до класу задач, що називаються екстремальними. Екстремальні задачі – це задачі пошуку екстремуму (мінімуму або максимуму) певної функції $L(x)$ на множині допустимих значень X , якій мають задовольняти точки x . Точка x в загальному випадку описується набором змінних x_i , тобто є вектором з координатами $(x_1, x_2, \dots, x_n)$. Точка $x(x_1, x_2, \dots, x_n)$, яка задовольняє обмеженням, що описують множину допустимих значень X , називається припустимим розв'язком екстремальної задачі. Функція $L(x)$ називається функцією цілі або цільовою функцією (аналогічно критерію оптимізації при пошуку оптимального значення x). Множина X також називається множиною припустимих розв'язків задачі. Припустимий розв'язок екстремальної задачі, на якому функція цілі досягає свого екстремуму, називається оптимальним розв'язком задачі або просто розв'язком.

Математична модель задачі лінійного програмування може бути записана в різному вигляді. Загальна задача лінійного програмування має наступний вигляд:

$$L(x) = \sum_{i=1}^n C_i x_i \rightarrow \min$$

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq b_1, \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \leq b_2, \\ \vdots \\ a_{k1}x_1 + a_{k2}x_2 + \dots + a_{kn}x_n \leq b_k, \\ a_{k+1,1}x_1 + a_{k+1,2}x_2 + \dots + a_{k+1,n}x_n = b_{k+1}, \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_m, \end{cases} \quad (4)$$

$$x_i \geq 0, \quad i = \overline{1, n}.$$

У середовищі MATLAB задачі лінійного програмування вирішуються за допомогою функції `linprog`. У даній роботі розглянемо її можливості.

Функція `linprog` вирішує задачу лінійного програмування у формі

$$\begin{aligned} C^T \cdot x &\rightarrow \inf, \\ A \cdot x &\leq b, \\ A1 \cdot x &= b1, \\ lb &\leq x \leq ub. \end{aligned} \quad (5)$$

Основними вхідними даними `linprog` є: вектор C коефіцієнтів цільової функції f , матриця обмежень-рівностей A , вектор правих частин обмежень-рівностей b , матриця обмежень-рівностей $A1$, вектор правих частин обмежень-рівностей $b1$, вектор lb , що обмежує план x знизу, вектор ub , що обмежує план x зверху. На виході функція `linprog` дає оптимальний план x завдання (1) та екстремальне значення цільової функції $fval$.

Розглянемо можливі варіанти вирішення задачі та знаходження

оптимального плану у системі комп'ютерної математики Matlab.

1. Задача має єдиний розв'язок.

$$f(x) = 4x_1 + 2x_2 + 3x_3 \rightarrow \inf,$$

$$2x_1 + x_2 + x_3 \geq 1, 3x_1 + x_2 - x_3 \geq -1, x_1 - x_2 + x_3 = 0,$$

$$0 \leq x_1 \leq 1, 0 \leq x_2 \leq 1, 0 \leq x_3 \leq 1.$$

Відповідний код у Matlab виглядає так:

```
clear all; close all;
```

clc % видаляються всі поточні змінні з пам'яті MATLAB, закриваються усі
графічні вікна, очищується екран консолі

```
C = (4 2 3); % задається вектор довжини три
```

```
D = (2 1 1; 3 1 -1); % рядки матриці поділяються крапкою з комою
```

```
B = (1 -1];
```

```
Aeq = (1 -1 1); beq = (0);
```

```
lb = zeros (3,1); % задається нульовий вектор довжини три
```

```
ub = (1 1 1);
```

```
f = C; A = -D; b = -B; % з'являються знаки -, оскільки обмеження-нерівності
```

$Dx \geq B$ приводяться до вигляду $-Dx \leq -B$

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

x

fval

Запустивши програму, отримаємо повідомлення

Optimization terminated.

x =

0.3333

0.3333

0

fval =

2

Додатково можна встановити початкове наближення x0:

```
[x,fval] = linprog(f,A,b,Aeq,beq,lb,ub,x0)
```

Якщо жодного з вхідних параметрів немає, на його місце слід поставити квадратні дужки [], за винятком випадку, коли це останній параметр у списку. Наприклад, якщо потрібно вирішити задачу без обмежень-рівностей, в якій не встановлено початкове наближення, то оператор виклику функції `linprog` буде виглядати так:

```
[x,fval] = linprog(f,A,b,[],[],lb,ub)
```

(Квадратні дужки наприкінці списку, що відповідають початковому наближенню, не ставляться.)

За допомогою вхідного параметра `options` встановлюються деякі додаткові налаштування, зокрема вибирається алгоритм рішення. MATLAB вирішує завдання лінійного програмування двома способами: алгоритмом внутрішньої точки (Large-Scale Algorithm) та варіантом симплекс-метода (Medium-Scale Algorithm). За замовчуванням використовується алгоритм внутрішньої точки. Щоб вибрати симплекс-метод, потрібно написати

```
options = optimoptions('linprog','Algorithm','dual-simplex')
```

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

Розберемося із вихідними даними. MATLAB дозволяє виводити інформацію про те, як завершилося рішення задачі. За це відповідає параметр `exitflag`. Якщо значення `exitflag` дорівнює 1, то знайдено розв'язання задачі, якщо дорівнює 0, то перевищено допустиме число ітерацій, якщо -2 — множина планів задачі порожня, якщо -3 — цільова функція не обмежена знизу на множині планів. Інтерпретація інших значень параметра `exitflag` наведена у MATLAB Help.

Для симплекс-метода допустима кількість ітерацій `MaxIteration` за замовчуванням у 10 разів більша за кількість змінних. Можна змінити значення, щоб встановити допустиме число ітерацій рівним, наприклад, 10, необхідно написати

```
options = optimoptions('linprog','Algorithm','dual-simplex','MaxIterations',10);
```

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

Якщо після виконання десятої ітерації рішення не буде знайдено, параметр

exitflag стане нульовим і на екрані з'явиться повідомлення

Maximum number of iterations exceeded; increase options.MaxIteration.

Параметр output містить інформацію про процес оптимізації, зокрема, число ітерацій (iterations) та алгоритм (algorithm). Інші поля параметра output описані у MATLAB Help. Запустимо з даними прикладу 1 наступну програму:

```
options = optimoptions('linprog','Algorithm','dual-simplex','MaxIterations',10);
```

```
[x,fval,exitflag,output] = linprog(f,A,b,Aeq,beq,lb,ub,options);
```

```
exitflag
```

```
output.iterations
```

```
output.algorithm
```

На виході отримаємо: Optimization terminated.

```
exitflag =
```

```
1
```

```
ans =
```

```
2
```

```
ans =
```

```
'dual-simplex'
```

Це означає, що симплекс-метод успішно завершив роботу, для знаходження рішення знадобилося дві ітерації.

2. При розв'язанні задачі лінійного програмування можливі три виходи з процесу: знайдено розв'язання задачі, множина планів порожня, цільова функція не обмежена знизу на множині планів. Продемонструємо ці варіанти на прикладах.

Вирішимо в MATLAB задачу лінійного програмування

$$f(x) = 4x_1 + x_2 \rightarrow \inf,$$
$$x_1 + x_2 \geq 2, x_1 - x_2 \geq 1, x_1 \geq 0, x_2 \geq 0.$$

Відповідна програма виглядатиме так:

```
clear all; close all; clc
```

```
C = (4 1); D = (1 1; 1 -1); B = (2 1);
```

```
lb = zeros (2,1); f = C; A = -D; b = -B;
```



```
options = optimoptions('linprog','Algorithm','dual-simplex');
```

```
[x,fval] = linprog(f,A,b,[],[],lb,[],options);
```

Optimal solution found.

```
[x,fval,exitflag] = linprog(f,A,b,[],[],lb,[],options);
```

Optimal solution found.

```
x =
```

```
1.5000
```

```
0.5000
```

```
fval =
```

```
6.5000
```

```
exitflag =
```

```
1
```

```
[x,fval,exitflag,output] = linprog(f,A,b,[],[],lb,[],options)
```

Optimal solution found.

```
x =
```

```
1.5000
```

```
0.5000
```

```
fval =
```

```
6.5000
```

```
exitflag =
```

```
1
```

```
output =
```

```
struct with fields:
```

```
iterations: 2
```

```
constrviolation: 0
```

```
message: 'Optimal solution found.'
```

```
algorithm: 'dual-simplex'
```

```
firstorderopt: 0
```

```
options = optimoptions('linprog','Algorithm','interior-point');
```

```
[x,fval,exitflag,output] = linprog(f,A,b,[],[],lb,[],options)
```

Minimum found that satisfies the constraints.

Optimization completed because the objective function is non-decreasing in feasible directions, to within the function tolerance, and constraints are satisfied to within the constraint tolerance.

x =

1.5000

0.5000

fval =

6.5000

exitflag =

1

output =

struct with fields:

iterations: 3

message: 'Minimum found that satisfies the constraints. Optimization completed because the objective function is non-decreasing in feasible directions, to within the function tolerance, and constraints are satisfied to within the constraint tolerance.'

algorithm: 'interior-point'

constrviolation: 0

firstorderopt: 1.7671e-07

options = optimoptions('linprog','Algorithm','dual-simplex','MaxIterations',10);

[x,fval,exitflag,output] = linprog(f,A,b,[],[],lb,[],options)

Optimal solution found.

x =

1.5000

0.5000

fval =

6.5000

exitflag =

1

output =

struct with fields:

iterations: 2

constrviolation: 0

message: 'Optimal solution found.'

algorithm: 'dual-simplex'

firstorderopt: 0

3. Вирішимо в MATLAB задачу лінійного програмування

$$f(x) = x_1 + x_2 \rightarrow \inf,$$

$$-x_1 - x_2 \geq -1, x_1 + 4x_2 \geq 8, x_1 \geq 0, x_2 \geq 0.$$

Наведемо результат роботи відповідної програми:

$$C = (1 \ 1); D = (-1 \ -1; 1 \ 4); B = [-1 \ 8];$$

$$lb = [0 \ 0];$$

$$f = C; A = -D; b = -B;$$

$$\text{options} = \text{optimoptions}('linprog','Algorithm','dual-simplex');$$

$$[x, fval] = \text{linprog}(f, A, b, [], [], lb, [], \text{options});$$

No feasible solution found.

Linprog stopped because no point satisfies the constraints.

Exiting: The constraints are overly stringent; no feasible starting point found.

Множина планів завдання порожня.

4. Вирішимо в MATLAB задачу лінійного програмування

$$f(x) = -x_1 - 3x_2 \rightarrow \inf, 2x_1 - x_2 \geq 0, -x_1 + x_2 \geq -1, x_1 \geq 0, x_2 \geq 0 \quad (4)$$

Запустивши програму, яка вирішує завдання (4), отримаємо:

Exiting: The solution is unbounded and at infinity;

x =

1.0e+016*

1.0000

2.0000

fval =

-7.0000e+016

exitflag =

-3

Цільова функція завдання (4) не обмежена знизу на багатьох планах.

Таким чином, ми розглянули вирішення задач лінійного програмування у системі комп'ютерної математики Matlab, провівши дослідження для усіх можливих варіантів рішення.