Зміст

Розділ 1

Введення

Говорячи про «дослідження операції», ми маємо на увазі пошук рішення якоїсь проблеми, пов'язаної із реальним життям. Дослідження полягає в тому, щоби знайти якнайкращий спосіб представлення реального світу чисельно: у вигляді матриць, векторів або функціональних залежностей. Тоді, використовуючи математичний апарат, відповідь на задачу можна знайти із будь-якою точністю. Після цього її потрібно інтерпретувати вже у «людських» поняттях.

Мистецтво чисельного опису світу називається побудовою **математичної моделі**. Така потреба виникала настільки часто, що з отриманих результатів можна виділити загальні підходи та ідеї, і використовувати їх на інших, ще не досліджених проблемах (операціях).

Цей посібник надає приклади багатьох популярних задач, що можуть бути розв'язані з допомогою лінійного, нелінійного або динамічного програмування, а також аналіз теорії, що стоїть за ними. Автор сподівається, що це дозволить сформувати хороше розуміння дослідження операцій, а також набути інтуїцію, яку потім можна використовувати при розв'язуванні нових, невідомих досі проблем.

1.1 Побудова математичної моделі

Логічно, що з початком аналізу будь-якої задачі, варто виділити ті дані, які подаються як факт – тобто, на них неможливо вплинути, чи змінити будь-як. Це те, біля чого можна написати «дано», і далі подати перелік змінних з присвоєними їм значеннями. Це називається **параметрами операції**, або **некерованими змінними**.

Якщо якісь дані описують одну і ту саму характеристику різних предметів, цілком доцільно об'єднати їх у вектор. Надалі використовуватиметься така нотація: $\mathbf{a} = \langle a_1, \ldots, a_n \rangle$. В тих випадках, коли подані дані є двовимірними, чи відображають деякі залежності, можна використовувати матриці. Наприклад, якщо в умові сказано, що *відомо час перевезення вантажу між кожним із обласних центрів*, то їх можна пронумерувати, а дані помістити, скажімо, в матрицю $\mathbf{T}$, де кожен елемент t_{ij} позначатиме час перевезення з i-ого міста в j-те.

Опісля маємо з'ясувати, які ж дані в задачі залишились невідомими, і їх потрібно знайти – себто, визначити на основі відомих. Вони називаються **керувальними параметрами**, або ж **керованими змінними**. Присвоєння цим змінним конкретних значень і є вирішенням проблеми.

У будь-яких задачах, які читач міг бачити до цього, зазвичай потрібно було з'ясувати значення одного невідомого параметра (деякого x), або навіть кількох. При дослідженні операцій може так траплятись, що невідомою буде ціла матриця, або вектор.

Якщо існує таке рівняння, що явно пов'язувало б некеровані змінні з керованими, то розв'язок можна знайти **аналітично**, тобто, обчислити звичайними арифметичними операціями. Наприклад, якщо відомо, що *пропускна спроможність каналу передачі становить n Мб/с, і потрібно з'ясувати, скільки максимально x Мб інформації можна передати за t секунд*, то зрозуміло, що відповідь може бути знайдена з формули $x = tn$.

Однак бувають такі проблеми, для яких неможливо – або принаймні дуже складно – скласти будь-яке рівняння, тому що у них взаємодіє між собою велика кількість змінних, кожна з яких по-своєму впливає на оптимальність результату в цілому. Тоді розв'язування може нагадувати намагання збалансувати складну систему тягарців: як тільки потягнути за один, одразу ж починають рух інші. Відповідь на такі проблеми найчастіше можна знайти **алгоритмічно**, на кожному з кроків покращуючи оптимальність результату. Власне, саме такий спосіб і використовується найчастіше для дослідження операцій.

Для того, щоби розв'язати задачу алгоритмічно, прийнятою практикою є побудова (опис) деякої множини G, що містить всі можливі допустимі розв'язки задачі. Звичайно, перед цим потрібно визначити, в якому вигляді ми взагалі шукатимемо розв'зок. Скажімо, якщо треба знайти дві змінні $x_1, x_2 \in \mathbb{R}$, то розв'язком буде вектор $\mathbf{X} = \langle x_1, x_2 \rangle$, і множина розв'язку – $G \subset \mathbb{R}^2$. Іноді рішенням проблеми може бути матриця, чи складніші багатовимірні об'єкти.

Наостанок, для того, щоб оцінити оптимальність будь-якого розв'язку, прийнято вводити **цінову функцію** (інша назва – **коефіцієнт ефективності**), що встановлює чисельну оцінку якості $\varphi : G \to \mathbb{R}$. Формулювання функції обирається залежно від умови.

Наприклад, якщо невідомі змінні x_1, x_2 встановлюють кількість товару двох видів, яку виготовлятиме завод, а c_1, c_2 – їхню ціну відповідно, то коефіцієнт ефективності можна визначити як прибуток від всього товару загалом: $\varphi(\mathbf{X}) = c_1 x_1 + c_2 x_2$. Тоді знайти такий $\mathbf{X}$, що забезпечував би максимальний прибуток, означає знайти екстремум φ.

1.2 Класифікація моделей

Запропонована структура математичної моделі дозволяє описати будь-яку операцію чисельно, зводячи розв'язування до пошуку такого $\mathbf{X} \in G$, щоби досягнути мінімуму або максимуму φ. Зрозуміло, що розв'язувати такі проблеми перебором, щонайменш неефективно. На щастя, для цього існують спеціальні методи, проте вони гарантують знаходження адекватного розв'язку лише для якогось конкретного типу математичної моделі. Тому важливо розрізняти, які типи існують взагалі, і вміти визначити їх для моделі своєї задачі. Далі приведено класифікацію математичних моделей за характерною ознакою.

	Так	Ні
Рішення проблеми знаходиться розв'язком рівняння.	**аналітична**	**алгоритмічна**
Хоча б одна з величин є випадковою величиною.	**стохастична**	**детермінована**
Кожен з кроків алгоритму розв'язку покращує рішення, знайдене на попередньому.	**статична**	
На кожному з кроків алгоритму розв'язку приймається найоптимальніше рішення.		**динамічна**
Хоча б одна з керованих змінних може приймати тільки цілі значення.	**дискретна**	**недискретна**

Розділ 2

Лінійне програмуванння

Одночасно простою для розуміння та корисною на практиці є задача лінійного програмування (ЗПЛ). Задачу можна назвати лінійною тоді, коли цінова функція є функцією першого степеня (2.1), а область допустимих значень може бути описана лінійними рівностями або нерівностями (2.2).

$$\varphi(\mathbf{X}) = \sum_{j=1}^{n} c_j x_j = \mathbf{c}\mathbf{X}^{\mathrm{T}},\ \mathbf{c} = \langle c_1, \ldots, c_n \rangle \tag{2.1}$$

$$G = \left\{ \langle x_1, \ldots, x_n \rangle : \sum_{j=1}^{n} a_{ij} x_j \boxed{\leq, \geq \text{ або } =} b_i \ \forall i = \overline{1, m} \right\} \tag{2.2}$$

2.1 Типові проблеми

Приведені далі постановки задач є типовими. Вони отримали свою назву від проблеми, у вирішенні якої найкраще використовувати такі означення. Якщо їх проаналізувати, то можна отримати відпрацьований математичний апарат для розв'язання більш специфічних задач.

2.1.1 Максимізація прибутку

Фабрика виробляє продукцію n видів. Для неї потрібна певна кількість сировини m видів, при чому її є в обмеженій кількості. Відомо можливий прибуток від реалізації кожного з видів продукції. Потрібно визначити, скільки кілограм товару кожного виду варто виготовити, щоби отримати якнайбільший прибуток.

Якщо між двома некерованими змінними існує відношення, то доцільно описати його у вигляді матриці. Тому нехай в $\mathbf{A}$ міститимуться елементи a_{ij}, кожен з яких означатиме витрату сировини i-ого виду на виготовлення продукції j-ого виду. Ще одна некерована змінна $\mathbf{b} = \langle b_1, \ldots, b_m \rangle$ позначатиме кількість відповідної сировини, доступного на фабриці. Прибуток від реалізації можна визначити вектором $\mathbf{c} = \langle c_1, \ldots, c_n \rangle$.

Керованою змінною є вектор $\mathbf{X} = \langle x_1, \ldots, x_n \rangle$, кожен елемент якого позначає кількість продукції, яку має виготовити фабрика – ці дані варто змінювати для досягнення оптимальності.

Тоді умовою допустимості розв'язку буде перш за все виконання обмежень на доступну кількість сировини, та деякі очевидні твердження – наприклад, те, що ця кількість має бути невід'ємною (2.3).

$$\mathbf{X} \in G \Leftarrow \left\{ \begin{array}{l} \sum\limits_{j=1}^{n} a_{ij} x_j \leq b_i \ \forall i = \overline{1, m} \\ x_j \geq 0 \ \forall j = \overline{1, n} \end{array} \right\} \tag{2.3}$$

Питання для роздумів

Чи можна поставити умову так, щоби знайдені x_i обов'язково повинні були би бути цілими числами?

В означенні (2.3) формулювання $\sum_{j=1}^{n} a_{ij}x_j$ позначає суму використаної сировини певного типу на виготовлення ресурсів всіх можливих видів. Відповідно «$\leq b_i$» позначає, що цією сировини можна використати не більше певної кількості. Умова $\forall i$ означає, що така нерівність повинна виконуватись для всіх видів сировини.

Цінову функцію потрібно вводити так, щоби вона визначала прибутковість прийнятого рішення. Тоді оптимальним розв'язком буде такий, що гарантує досягнення її максимуму (2.4).

$$\varphi(\mathbf{X}) = \sum_{j=1}^{n} c_j x_j \to \max_{\mathbf{X}\in G} \tag{2.4}$$

Питання для роздумів

Якщо існує така задача, для якої потрібно вводити кілька цінових функцій, і оптимізовувати їх одночасно, то якою буде умова такої задачі? Чи існує якийсь універсальний спосіб поєднання кількох оцінок оптимальності в одну?

2.1.2 Мінімізація витрат

Розглянемо задачу, подібну тій, що поставлена у розділі 2.1.1. *Фабрика отримує сировину n видів, з якої виготовляє товар за m різними технологіями. Кожна з технологій потребує певну комбінацію сировини у різних кількостях. Разом з тим, кожна з технологій спричиняє в атмосферу певну кількість кілограм вуглецю. Для кожного виду сировини визначена* *доступна кількість, яку можна використати для виробництва. Потрібно оптимізувати виробництво так, щоби отримати якомога більше готової продукції, але водночас зменшити кількість шкідливих викидів.*

Що в цій задачі уже відомого? Найперше – відношення технології та сировини, яку вона потребує. Запишемо це в матрицю $\mathbf{A}$, в якій кожен елемент a_{ij} позначатиме витрату сировини i-ого типу за j-ою технологією. Викиди різних технологій можна зберігати у векторі $\mathbf{c} = \langle c_1, \ldots, c_m \rangle$, а кількість доступної сировини кожного виду – як $\mathbf{b} = \langle b_1, \ldots, b_n \rangle$.

Що потрібно знайти? Потрібно знайти кількість товару, яка буде виготовлена за j-ою технологією. Тому позначимо її через вектор $\mathbf{X} = \langle x_1, \ldots, x_m \rangle$. При пошуку значення різних x_j доведеться вирішити дилему: користуватись технологіями, які потребують меншу кількість сировини, але спричиняють більше викидів, чи навпаки?

Будь-який розв'язок $\mathbf{X}$ буде допустимим, якщо задовольнятиме дві умови: використано ресурсів не більше, ніж доступно взагалі, і жоден з x_j не є від'ємним. Тому ознака допустимості розв'язку є такою самою, як і (2.3) – ми використовуємо ті самі назви змінних, тому формула не втрачає свого змісту. Тепер формулювання $\sum_{j=1}^{n} a_{ij} x_j$ позначатиме кількість сировини певного виду, використаної загалом, за якими технологіями б не відбувалось виробництво.

Допоки розв'язок допустимий, він використовуватиме лише доступну кількість сировини. Однак різні розв'язки все ще спричиняють різний викид вуглецю, топу оптимальтність кожного з них визначатимемо саме за цим параметром. Цікаво, що цінова функція знову така сама, як і в (2.4), проте на цей раз її доцільно мінімізувати.

$$\varphi(\mathbf{X}) = \mathbf{c}\mathbf{X}^{\mathrm{T}} \to \min_{\mathbf{X} \in G} \tag{2.5}$$

Тепер, після ознайомленнями з двома задачами, важливо розуміти, що не існує жодних правил, як варто складати формулювання ОДР чи цінової функції, так само як і неможливо передбачити всі можливі «типові задачі». Формулювання будь-якої математичної моделі залежить винятково від області дослідження та винахідливості дослідника.

2.1.3 План перевезень

Досі ми розглядали задачі, де невідомою керованою змінною був вектор – тобто, кількість чогось різних видів, або оптимальна конфігурація різних характеристик. Чи можливо тепер за допомогою лінійного програмування розв'язати якусь проблему, що встановлюватиме конфігурацію відношення? Тобто, цього разу невідоме $\mathbf{X}$ буде двовимірним – матрицею.

Компанія має у своєму розпорядженні n зерносховищ та m постачальників зерна. Кожне зерносховище може зберігати певну кількість тон зерна, купленого у різних постачальників. З ними укладено контракт, за яким компанія зобов'язується викупити щонайменш певну кількість тон за раз. Відомі затрати на перевезення зерна від кожного постачальника до кожного сховища – скажімо, час, відстань, пальне тощо. Потрібно визначити, скільки зерна між різними постачальниками та відповідними сховищами доцільно перевезти, щоби отримати якомога менші витрати, та перевезти якомога більше вантажу.

Отже, умову проблеми можна зобразити у вигляді графа, що на Рис. 2.1. Зрозуміло, що некерованими змінними є $\mathbf{a}$, $\mathbf{b}$ та $\mathbf{C}$. Тоді розв'язком буде матриця $\mathbf{X}$, кожен елемент x_{ij} якої задаватиме «оптимальну» кіль-

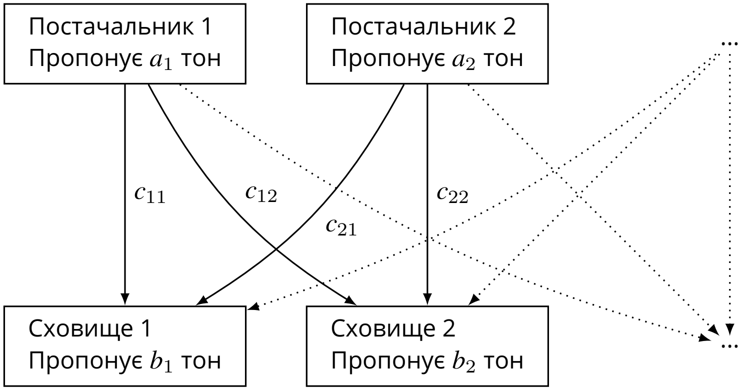

Рис. 2.1: Кожен з постачальників пропонує $\mathbf{a} = \langle a_1, \ldots, a_m \rangle$ тон, а кожне зі сховищ може зберігати до $\mathbf{b} = \langle b_1, \ldots, b_n \rangle$. Затрати на перевезення між ними задаються матрицею $\mathbf{C} \in \mathbb{R}^{m \times n}$.

кість тон, яку варто перевезти між i-им постачальником та j-им сховищем. Саме така конфігурація, що буде задана цією матрицею і гарантуватиме найменші витрати та якнайбільше заповнення сховищ.

Ця проблема є класичним випадком транспортної задачі, і для її розв'язку існують набагато ефективніші методи, про які можна дізнатись у наступних розділах. Попри це, розв'язок саме лінійним програмуванням є досить цікавим.

Тепер ОДР повинно мати такі обмеження:

- потрібно вивезти весь товар, що надають постачальники;
- до кожного з сховищ не можна привезти більше, ніж воно може вмістити.

Тоді умову допустимості розв'язку можна записати як 2.6.

$$\mathbf{X} \in G \Leftarrow \left\{ \begin{array}{l} \sum_{i=1}^{m} x_{ij} = a_i \ \forall j = \overline{1, n} \\ \sum_{j=1}^{n} x_{ij} \leq b_j \ \forall i = \overline{1, m} \end{array} \right\} \tag{2.6}$$

Тоді для того, щоби порахувати затрати при поточному плану перевезення, достатньо перемножити значення c_{ij} з матриці затрат на значення x_{ij}. Оптимальним буде той розв'язок, у якому це значення найменше, тому вводимо цінову функцію, і вимагаємо її мінімізації 2.7.

$$\varphi(\mathbf{X}) = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \rightarrow \min_{\mathbf{X} \in G} \tag{2.7}$$

Питання для роздумів

Нехай матриця $\mathbf{C}$ відображає не затрати на перевезення, а максимальну пропускну здатність маршруту. Скажімо, в день по дорозі між двома пунктами можна перевезти не більше, ніж c_{ij} тон вантажу. Потрібно знайти таке завантаження $\mathbf{X}$ різних доріг, щоби перевезти якомога більше вантажу. Допускається ситуація, коли не весь вантаж буде вивезено від постачальників. Як сформулювати ОДР та цінову функцію?

2.2 Проблеми оптимального вибору

Припустимо таку ситуацію, коли ми маємо набір різних рішень, з яких складається розв'язок проблеми. Наприклад, *складне обчислення можна розпаралелити на n процесорів, кожен з яких має свою продуктивність операцій/с та потужність у ватах. Скільки і які процесори потрібно* *увімкнути, щоби продуктивності вистачило на виконання деякого обчислення у вказаний час, але спожити щонайменше енергії?* Тоді набір всіх можливих рішень

$$P = \{\text{увімкнути процесор 1}, \text{увімкнути процесор 2}, \ldots, \\ \text{увімкнути процесор } n\}.$$

Відповідно, будь-який розв'язок задачі буде лише підмножиною P. Однак такими формулюваннями оперувати дуже не дуже зручно, тому краще ввести характеристичний вектор $\mathbf{X}$, кожен елемент x_i якого позначатиме, чи прийняте деяке i-те можливе рішення. Так, якщо $x_i = 0$, то i-ий процесор варто вимкнути, але якщо $x_i = 1$ – то навпаки.

Формально такий випадок можна записати як $\mathbf{X} \in \{0, 1\}^n$, або простіше – $\forall i : x_i \in \{0, 1\}$.

Розділ 3

Нелінійне програмування

Розділ 4

Динамічне програмування