

Туржанська Оксана Степанівна – асистент кафедри математики і методики навчання математики Вінницького державного педагогічного університету ім. Михайла Коцюбинського, кандидат педагогічних наук.

Коло наукових інтересів: моніторинг якості підготовки фахівців.

ОПТИМІЗАЦІЯ НА ГРАФАХ З СИСТЕМОЮ “МАТНЕМАТИКА”

Тарас КОБИЛЬНИК, Уляна КОГУТ

У статті проаналізовано різні підходи використання системи комп'ютерної математики Mathematica до розв'язування деяких оптимізаційних задач з використанням елементів теорії графів.

The article analyzes the different approaches of using computer mathematics system Mathematica to solve some optimization problems using graphs theory.

Постановка проблеми та аналіз останніх досліджень. Значну кількість оптимізаційних задач можна розв'язати за допомогою засобів теорії графів. Питанням, пов'язаним з використанням графів для розв'язування оптимізаційних задач, присвячені роботи [1; 4; 6]. М.Н. Кірсанов [3] розглядає можливості використання системи комп'ютерної математики (СКМ) Maple для розв'язування задач з теорії графів.

З елементами теорії графів студенти знайомляться у курсі «Дискретна математика». Крім того, студенти поглиблюють свої знання з теорії графів при вивченні дисципліни «Алгоритми і структури даних». Окремі алгоритми розв'язування оптимізаційних задач на графах вивчають у курсі «Дослідження операцій та теорія ігор». Саме розв'язуванню таких задач використанням СКМ Mathematica, що за дослідженням С.Стейхауса (S. Steihaus) [7] є кращою в середньому за всіма категоріями порівняння, в тому числі і за математичними характеристиками, присвячена дана стаття.

Мета статті: аналіз можливостей використання СКМ Mathematica при розв'язуванні оптимізаційних задач на графах.

Виклад основного матеріалу. Оптимізаційні алгоритми на графах зручно використовувати для розв'язування таких задач як побудова каркасу мінімального (максимальної) ваги; пошук найкоротшого шляху; визначення максимального потоку; мінімізація вартості потоку в мережі з обмеженою пропускну здатністю; задачу комівояжера.

Безпосереднє розв'язування оптимізаційних задач за допомогою засобів теорії графів є досить трудомістким та складним процесом. На даний момент існують програмні засоби, які дозволяють розв'язувати лише спеціалізовані оптимізаційні задачі, що унеможлиблює їх використання для розв'язання широкого кола оптимізаційних задач.

У системі Mathematica [2] для розв'язування задач з теорії графів зручно використовувати пакет розширення Combinatorica.

Каркас мінімальної (максимальної) ваги. Ця задача виникає при проектуванні ліній електропередач, трубопроводів, доріг тощо, коли вимагається з'єднати центри деякою системою каналів зв'язку таким чином, щоб будь-які два центри були з'єднані або безпосередньо, або через інші центри і канали, і щоб загальна довжина була найменшою. Задані центри можна вважати вершинами графу, а канали зв'язку – ребрами з відповідними вагами. Для розв'язування такої задачі розроблені алгоритми Краскала та Прима, що застосовуються до довільного зв'язного графу.

За функцією MinimumSpanningTree[G] будується каркас мінімальної ваги для графа G. У цій функції використовується алгоритм Краскала.

Приклад 1. Телевізійна компанія планує під'єднати до кабельної мережі п'ять нових районів. На рис. 1 наведено структуру мережі і відстані між районами та телецентром. Необхідно спланувати найбільш економічну кабельну мережу. [6, с.246]

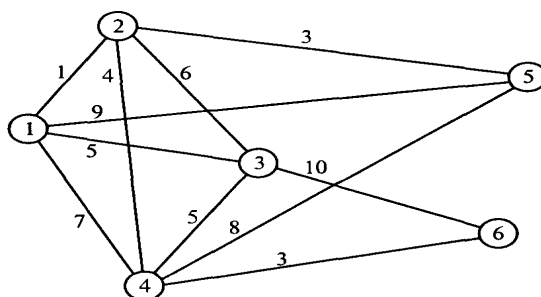
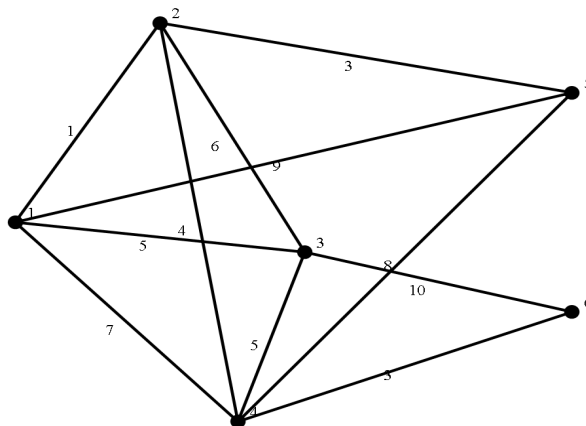


Рис. 1

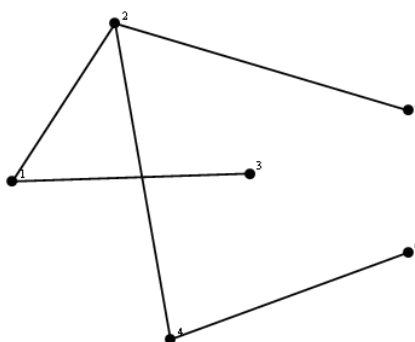
Розв'язування. Перш за все звернемося до функцій пакету розширень Combinatorica. Далі за функцією FromUnorderedPairs створимо неорієнтований граф з заданням ваг ребер (SetEdgeWeights), вказавши номери вершин (SetVertexLabels). За функцією ShowGraph будемо створений граф. За функцією MinimumSpanningTree будемо каркас мінімальної ваги. Для графічної візуалізації використовуємо команду ShowGraph. Далі за виразом Apply[Plus, GetEdgeWeights[g, Edges[z1]]] обчислюємо мінімальну вагу (рівна 16) створеного каркасу. У системі Mathematica наведений вище опис можна реалізувати таким чином:

```
<< Combinatorica`
```

```
v = {{-1.5, 0}, {-0.85, 1}, {-0.2, -0.15}, {-0.5, -1},  
      {1, 0.65}, {1, -0.45}};  
G = FromUnorderedPairs[{{1, 2}, {1, 5}, {1, 3}, {1, 4},  
                        {2, 5}, {2, 3}, {2, 4}, {3, 4}, {3, 6}, {4, 5}, {4, 6}}, v];  
G = SetEdgeWeights[G, {1, 9, 5, 7, 3, 6, 4, 5, 10, 8, 3}];  
G = SetEdgeLabels[G, {1, 9, 5, 7, 3, 6, 4, 5, 10, 8, 3}];  
G = SetVertexLabels[G, {1, 2, 3, 4, 5, 6}];  
ShowGraph[G, EdgeLabel -> GetEdgeWeights[G], VertexLabel -> True]
```



```
z1 = MinimumSpanningTree[g]; ShowGraph[z1]  
Apply[Plus, GetEdgeWeights[g, Edges[z1]]]
```



Знаходження найкоротших шляхів. Ця задача полягає у знаходженні у транспортній мережі найкоротшого шляху між заданим пунктом і пунктом призначення. Одним з прикладів такої задачі може бути завдання про знаходження найкоротшого шляху між двома вузлами в мережі.

Приклад 2. Головоломка про три бідони [6]. Восьмилітровий бідон заповнений водою, а два бідони об'ємом 5 і 3 літри порожні. Необхідно розділити 8 літрів води на дві рівні частини, використовуючи тільки ці бідони. Яку мінімальну кількість переливань з бідона в бідон потрібно виконати, щоб досягти бажаного результату. Розв'язати цю задачу як задачу про знаходження найкоротшого шляху.

У цій моделі кожна вершина графу буде відповідати об'ємам води у 8-, 5- та 3-літрових бідонах. Початковою вершиною графу буде $(8, 0, 0)$, а кінцевим – $(4, 4, 0)$. Нова вершина графу отримується з попереднього при одноразовому переливанні води з одного бідона в інший.

Разом зі студентами викладач аналізує варіанти отримання бажаного результату, при цьому можливі варіанти зображаються у вигляді графу (див. рис.2). Після побудови графу задачі студенти, використовуючи, наприклад алгоритм Дейкстри, відшуковують розв'язок задачі. Крім того, студентам пропонується розв'язати цю ж задачу з допомогою функцій пакету Combinatorica системи Mathematica.

Для досягнення бажаного (за мінімальну кількість переливань) результату потрібно сім переливань з бідона в бідон (розв'язок, показаний у нижній частині рис. 2).

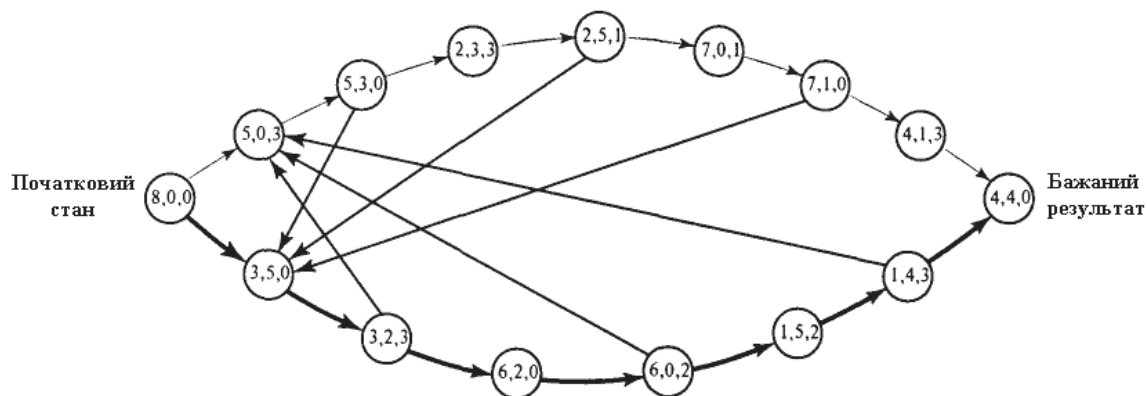


Рис. 2. Граф задачі „Головоломка про три бідони”

Таким чином, студенти розв'язують задачу з теми заняття і разом з тим знімається надлишкове навантаження та уникається зниження зацікавленості; студенти отримують певне зацікавлення – на прикладі показано застосування теоретичного матеріалу.

Приклад 3. Заданий орієнтований граф (рис.3). Знайти найкоротший шлях з вершини 1 до вершини 6.

Розв'язування. За функцією FromOrderedPairs задаємо орієнтований граф з вказанням номерів вершин та ваг ребер.

```
v = {{-1, -1}, {0, -1}, {1, -1}, {-1, 1}, {0, 1}, {1, 1}};
G = FromOrderedPairs[{{1, 2}, {1, 4}, {1, 5}, {2, 3}, {2, 5},
  {3, 6}, {4, 5}, {5, 6}}, v];
G = SetEdgeWeights[G, {11, 12, 26, 15, 16, 20, 13, 14}];
G = SetEdgeLabels[G, {11, 12, 26, 15, 16, 20, 13, 14}];
G = SetVertexLabels[G, {1, 2, 3, 4, 5, 6}];
ShowGraph[G, EdgeLabel -> GetEdgeWeights[G], VertexLabel -> True]
```

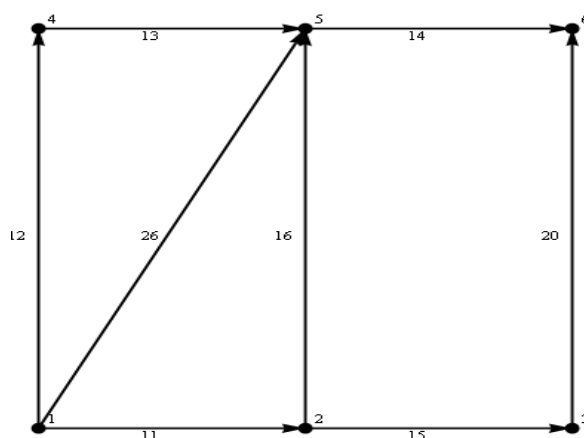


Рис. 3.

За функцією $\text{Dijkstra}[G, 1][[2, 6]]$ обчислюємо мінімальну відстань (дорівнює 39) від вершини 1 до вершини 6 та шлях (за функцією $\text{ShortestPath}[G, 1, 6]$), що їй відповідає (проходить через вершини 1, 4, 5, 6 в заданому порядку):

$\{\text{Dijkstra}[G, 1][[2, 6]], \text{ShortestPath}[G, 1, 6]\}$

$\{39, \{1, 4, 5, 6\}\}$

Приклад 4. При вивченні питання про знаходження найкоротшого шляху в мережі студентам можна запропонувати таке завдання. Студент щоденно (крім вихідних) ходить до університету. Він визначив найкоротший шлях з дому до університету. Проте на цьому шляху він зустрічає друзів і з ними кілька хвилин спілкується. Таким чином, найкоротший шлях виявився не найшвидшим. Тому студент хоче визначити новий маршрут, на якому він би мав найбільшу ймовірність не зустріти своїх друзів. Схема мережі доріг, якими студент може потрапити з дому до університету показана на рис. 4. На цій же схемі наведені ймовірності *не зустріти друзів* для кожного сегмента мережі доріг. Ймовірність не зустріти друзів дорівнює добутку ймовірностей на кожному сегменті вибраного шляху. Студенту необхідно розв'язати задачу вибору маршруту, який би максимізував ймовірність не зустріти друзів.

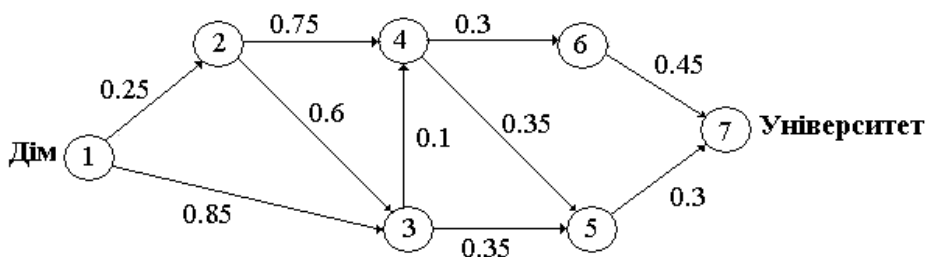


Рис. 4. Схема мережі доріг з дому до університету

Головне завдання студента полягає у формулюванні даної задачі як задачі на знаходження найкоротшого шляху. Це можна зробити, замінивши ймовірності логарифмами ймовірностей. Тоді добуток ймовірностей перетвориться у суму логарифмів ймовірностей: $p_{ik} = p_1 \cdot p_2 \cdot \dots \cdot p_k$ – ймовірність не зустріти друзів на маршруті $1 \rightarrow 2 \rightarrow \dots \rightarrow k$, тоді $\ln p_{ik} = \ln p_1 + \ln p_2 + \dots + \ln p_k$. Задача максимізації ймовірності p_{ik} еквівалентна задачі максимізації величини $\ln p_{ik}$. Оскільки $\ln p_{ik} \leq 0$, то задача максимізації величини $\ln p_{ik}$ еквівалентна задачі мінімізації $-\ln p_{ik}$. Таким чином, замінивши ймовірності p_k на величини $-\ln p_k$ отримаємо мережу (рис.5), до якої можна застосовувати алгоритм знаходження найкоротшого шляху.

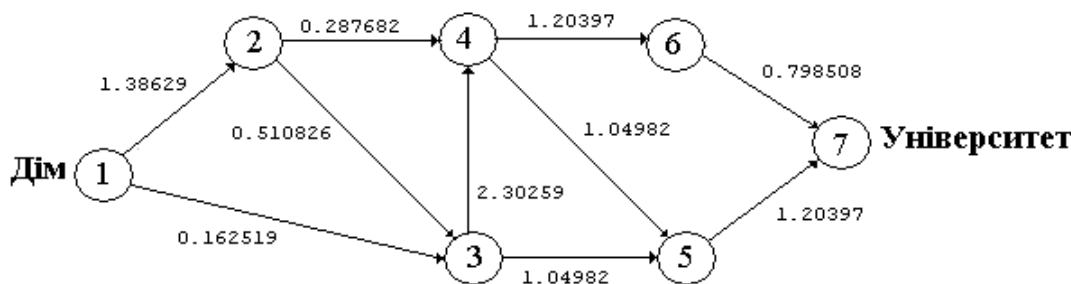


Рис. 5. Мережна модель для задачі знаходження найкоротшого шляху

Знаходження найкоротшого шляху студентам можна запропонувати в одній з СКМ, зокрема Mathematica, використавши функцію Dijkstra. Обчислений найкоротший шлях для отриманої мережі: $1 \rightarrow 3 \rightarrow 5 \rightarrow 7$ з відповідною довжиною шляху 2.416309 ($= -\ln p_{17}$). Таким чином, максимальна ймовірність не зустріти друзів дорівнює $p_{17} \approx 0.089$.

Задача комівояжера. Цікавою для аналізу є задача комівояжера [5] – одна з базових задач комбінаторної оптимізації, що має широке прикладне застосування. Існує небагато алгоритмів, що забезпечують одержання якісних розв'язків задачі комівояжера.

Нехай $P = (1, 2, \dots, n)$ – скінченна множина точок, $c_{ij} \geq 0$ – відстань (вартість) від точки i до точки j , $c = (c_{ij})_{n \times n}$ – матриця відстаней (вартостей). Маршрут z комівояжера – це довільна перестановка точок з P , $z = (i_1, i_2, \dots, i_n)$. Довжина маршруту z – сума цих відповідних елементів матриці (c_{ij}) :

$$l(z) = \sum_{k=1}^n c_{i_k i_{k+1}}, \quad i_{n+1} = i_1.$$

Нехай Z – множина всіх маршрутів. Потрібно знайти маршрут $z_0 \in Z$ такий, що $l(z_0) = \min l(z)$, $z \in Z$.

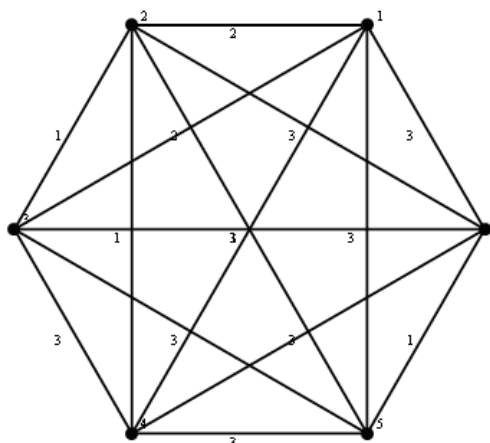
Наведемо приклад розв'язування симетричної задачі комівояжера ($c_{ij} = c_{ji}$) для такої матриці:

$$\begin{pmatrix} \infty & 2 & 2 & 3 & 3 & 3 \\ 2 & \infty & 1 & 1 & 1 & 3 \\ 2 & 1 & \infty & 3 & 3 & 3 \\ 3 & 1 & 3 & \infty & 3 & 3 \\ 3 & 1 & 3 & 3 & \infty & 1 \\ 3 & 3 & 3 & 3 & 1 & \infty \end{pmatrix}.$$

Побудуємо неорієнтований граф, що визначається заданою матрицею:

```
<< Combinatorica`
```

```
g = FromUnorderedPairs[{{1, 2}, {1, 3}, {1, 4}, {1, 5}, {1, 6}, {2, 3},
  {2, 4}, {2, 5}, {2, 6}, {3, 4}, {3, 5}, {3, 6}, {4, 5}, {4, 6}, {5, 6}}];
g = SetEdgeWeights[g, {2, 2, 3, 3, 3, 1, 1, 1, 3, 3, 3, 3, 3, 3, 1}];
g = SetEdgeLabels[g, {2, 2, 3, 3, 3, 1, 1, 1, 3, 3, 3, 3, 3, 3, 1}];
g = SetVertexLabels[g, {1, 2, 3, 4, 5, 6}];
ShowGraph[g]
```



За функцію `TravelingSalesman[g]` визначається оптимальний шлях комівояжера в неорієнтованому графі `g`:

`TravelingSalesman[g]`

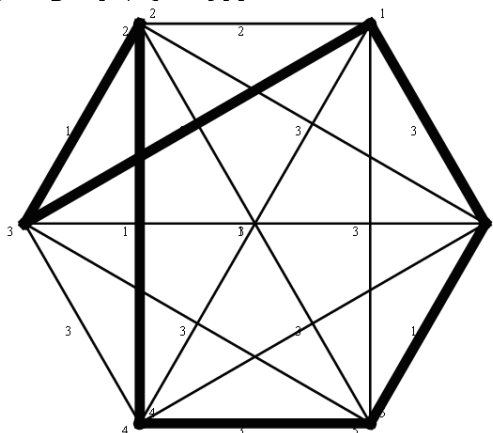
`{1, 3, 2, 4, 5, 6, 1}`

Цей результат можна інтерпретувати таким чином: оптимальним розв'язком є $z_0 = (1, 3, 2, 4, 5, 6, 1)$.

Виділимо оптимальний маршрут (товста лінія) за допомогою функції `Highlight`:

`h = SetGraphOptions[g, {{1, 6, VertexNumberPosition -> UpperRight}}];`

`ShowLabeledGraph[Highlight[h, {tur}]]`



Ціна оптимального маршруту обчислюється за функцією `CostOfPath`:

`CostOfPath[h, s]`

11

Висновок.

Можливості використання системи Mathematica для розв'язування задач з теорії графів значні. Розглянуті у статті методи та алгоритми надають зручний інструмент для розв'язування деяких оптимізаційних задач з використанням графів. Такий підхід сприяє виробленню у студентів вмінню моделювати та розв'язувати оптимізаційні задачі на графах з використанням СКМ Mathematica. Студент, використовуючи СКМ Mathematica, розв'язує поставлену перед ним задачу, і таким чином, у нього не виникає психологічного бар'єру у застосуванні математичного апарату, а також усвідомлює, який матеріал треба повторити (або вивчити). Розв'язування оптимізаційних задач на графах з використанням СКМ надає знанням і вмінням студентів практично значущого характеру.

БІБЛІОГРАФІЯ

1. Воденин Д. Р. Оптимизационные задачи на графах: Учебно-методическое пособ. для студ.экон.эфак./ Д.Р. Воеводин. – Ульяновск: УлГУ.Мех.-мат.фак, 1999. – 72 с.
2. Дьяконов В. П. МАТЕМАТИКА 5.1/5.2/6.0. Программирование и математические вычисления / Владимир Павлович Дьяконов. – М. : ДМК Пресс, 2006. – 576 с.
3. Кирсанов М. Н. Графы в Maple / Кирсанов Михаил Николаевич. – М : Физматлит, 2007. – 168 с.
4. Новиков Ф.А. Дискретная математика для программистов / Ф.А. Новиков. – СПб.: Питер, 2005. – 364 с.
5. Сигал И. Х. Введение в прикладное дискретное программирование: модели и вычислительные алгоритмы: учеб.пособие / И. Х. Сигал, Иванова А. П. – [изд. 2-е, испр.]. – М. : ФИЗМАТЛИТ, 2003. – 240 с.
6. Таха Х. А. Введение в исследование операций / Хемди А. Таха; пер. с англ. – [7-е издание]. – М. : Издательский дом „Вильямс”, 2005. – 912 с.
7. Steinhaus S. Comparison of Mathematical Programs for Data Analysis (Edition 5.03) [Електронний ресурс] – Munchen/Germany. – 64 p. – Режим доступа : <http://www.scientificweb.de/ncrunch/>.

ВІДОМОСТІ ПРО АВТОРІВ

Кобильник Тарас Петрович – доцент кафедри інформатики та обчислювальної математики Дрогобицького державного педагогічного університету імені Івана Франка.

Когут Уляна Петрівна – аспірант Інституту інформаційних технологій і засобів навчання НАПН України м.Київ, викладач кафедри інформатики та обчислювальної математики Дрогобицького державного педагогічного університету імені Івана Франка.

Коло наукових інтересів: використання систем комп'ютерної математики при навчанні інформатичних дисциплін.

ПІДГОТОВКА ВЧИТЕЛІВ ПРИРОДНИЧИХ ДИСЦИПЛІН ДО РОЗРОБКИ І ВПРОВАДЖЕННЯ ВІДКРИТИХ ЗАДАЧ

Євгенія КОСТЕНКО

У статті розглянуто сучасний підхід до організації методичної роботи школи, спрямованої на стимуляцію впровадження у педагогічну практику вчителів природничих наук «відкритих завдань», на матеріалах науково-методичної літератури відстеженні існуючі підходи, запропонована схема такої роботи.

The paper examines the up-to date approach to organization of teaching technique in school, directed at stimulation of incorporation of 'open tasks' into educational work of the natural science teachers, using materials of methodological literature, following up the existing approaches to such activity.

Класичні педагогічні прийоми й традиційні методики викладання, безумовно, є наріжним каменем сучасної середньої освіти. Але більшість учителів зіштовхується із проблемою зниженої навчальної мотивації школярів, відсутністю інтересу до пізнавальної діяльності, з байдужістю або, взагалі, неприйняттям того, що викладається в школі. Методики викладання, які працювали в умовах тоталітарної системи, дуже часто не сприймаються сучасним поколінням школярів, вихованим в умовах, що змінилися. І це закономірно, адже актуальною для сучасних випускників є майбутня успішність, заможність у всіх змістах цього слова, вміння вирішувати складні багатофакторні життєві завдання, а не певний обсяг знань, який намагаються передати їм на уроках. Це висуває принципово нові вимоги перед учителем – сформувати у школярів компетенції, як уміння застосовувати знання для прийняття рішення в нестандартній ситуації [1, с.166], навчити дитину мислити та виховати з неї успішну людину. Тому важливою складовою професіоналізму вчителя стає готовність до створення і впровадження інновацій, а також до коректної експериментальної перевірки інноваційних ідей.

Основні методологічні й теоретичні положення інноваційної педагогічної діяльності розглядалися у роботах О.Арламова, К.Ангеловські, І.Беха, М.Бургіна, Ю.Гільбуха, І.Дичківської, Д.Мазоха, Н.Опанасенко, А.Ніколса, С.Полякова, М.Поташника, Г.Селевка, Н.Юсуфбекової.

Численні дослідження психологів, дидактів і методистів (Г.Балла, П.Гальперіна, В.Давидова, В.Загвязинського, Ю.Кулюткіна, І.Лернера, А.Матюшкина, М.Махмутова, В.Оконя, З.Слепкань, Л.Фрідмана й ін.) доводять, що створення сприятливих умов для навчання і розвитку особистості пов'язане з веденням в освітній процес навчально-пізнавальних задач діяльнісно-ціннісного змісту. Тому вирішення проблеми практичної реалізації задачного підходу в умовах інноваційного навчання є актуальним і важливим для сучасної освіти.