

АДАПТИВНИЙ АЛГОРИТМ ФОРМУВАННЯ МАТЕМАТИЧНОЇ МОДЕЛІ ГАЗОТРАНСПОРТНОЇ СИСТЕМИ

© Притула Н., 2013

Розглядається задача знаходження режимних параметрів роботи газотранспортної системи, яка зводиться до розв'язання системи рівнянь у частинних похідних методом скінченних елементів. На стійкість знаходження розв'язку суттєво впливає крок за часовою координатою. Запропоновано адаптивний алгоритм зменшення впливу величини кроків на стійкість розв'язку. Крім цього, запропоновано алгоритм зменшення часу розв'язування режимних задач способом редагування технологічних схем – моделі структури системи транспортування газу.

Ключові слова: газотранспортна система, адаптивний алгоритм, стійкість, числовий експеримент, технологічна схема.

The problem of finding of operation modes parameters of a gas transmission system which is reduced to solving of partial differential equations set by a finite element method is considered. Fastness of a decision finding is influenced significantly by a step of hour coordinate. The adaptive algorithm of influence reduction of steps size on a decision fastness is offered. The algorithm of solving time reduction of operation modes tasks by way of technological schemes editing – models of gas transportation system structure is besides offered.

Key words: gas transmission system, adaptive algorithm, fastness, numerical experiment, technological scheme.

Вступ

Математична модель газотранспортної системи формується на основі її технологічної схеми. Структурні властивості технологічної схеми впливають як на розмірність системи рівнянь – моделі системи, так і на складність її розв'язування [1–6]. Проведені числові експерименти показали, що деякі перетворення технологічної схеми як граф-схеми забезпечують більшу стійкість методу і зменшують час розв'язування відповідних систем. Основними такими операціями над графами є об'єднання ребер і підграфів та їх ототожнення з окремими ребрами і вершинами. Важлива послідовність проведення операцій. Суттєві обмеження на операції і їх послідовність накладають існуючі граничні умови в режимних задачах. Процес розв'язування задач потребує максимальної автоматизації. Вона, своєю чергою, вимагає побудови відповідних алгоритмів, які б забезпечили, перед моделюванням газодинамічних процесів, повну автоматизацію процесу редагування вихідних технологічних схем.

1. Система редагування технологічних схем

У програмному комплексі схема ГТС представлена у вигляді графа $G = (V, E)$, де V – множина вершин (вузлів), E – множина ребер. Ребрами представляються об'єкти, які мають протяжність у просторі (труби, крани, компресорні станції (КС), редуктори), вершинами – всі інші об'єкти (стики труб, джерела, відбори, свічі тощо).

Для зменшення розмірності системи лінійних рівнянь та збільшення стійкості методу її розв'язування зменшимо кількість графічних об'єктів у технологічній схемі газотранспортної системи (граф $G(X, Y)$). Для цього запропоновано такий алгоритм:

а) вершини, інцидентні ребрам (дугам) графу технологічної схеми (довжина ребер яких менша за задану L), ототожнюються, тобто ребру $(a_i, a_j) \in Y$ довжиною $|a_i, a_j| \leq L$ графу $G(X, Y)$ ставлять у відповідність одну із вершин a_i чи a_j ;

б) маршруту (a_1, a_2, \dots, a_n) , в графі $G(X, Y)$ якого всі вершини, можливо, за винятком a_1 та a_2 степеня два, відповідає ребро (a_1, a_n) , якщо всі ребра $(a_i, a_{i+1}) (i=1, 2, \dots, n-1)$ мають однаковий внутрішній діаметр і той самий номінальний тиск газу;

в) ребро, паралельне до ребра (a_i, a_j) , заміняє на маршрут (a_i, a, a_j) так, щоб $|a_i, a_j| = |a_i, a| + |a, a_j|$;

г) вершині a , яка отримана в процесі ототожнення вершин a_i з граничними умовами $Q_i(t)$, присвоюється гранична умова $\sum_i Q_i$;

д) ребро, яке задовольняє умову а, та одна із його вершин з нульовою граничною умовою на витрату ототожнюється з іншою вершиною того ж ребра, але без граничної умови;

е) не ототожнює вершин із граничними умовами на різні параметри.

Розрізняються відбори (надходження) газу з відводів (у відводи) та безпосередньо з трубопроводу (у трубопровід). Сказане не відповідає відводам з нульовою довжиною, оскільки вказані випадки по-різному входять в систему рівнянь.

У разі зміни стану об'єкта типу ребро у вершинах, інцидентних цьому ребру, формуються граничні умови, які враховують модель і технологічний процес роботи об'єкта.

2. Алгоритм побудови розрахункових схем

Параметри нестационарних динамічних процесів знаходять методом скінченних елементів із системи нелінійних диференціальних рівнянь в частинних похідних. Кожне ребро граф-схеми об'єкта моделювання містить певну кількість (яка залежить від довжини ребра) елементів розбиття (вузлів), для кожного з яких будується система рівнянь із двох рівнянь згідно із симплекс-методом з чотирикутними елементами. Тому важливим критерієм швидкодії розв'язування є підбір мінімальної кількості елементів розбиття ребер, щоб отримати меншу розмірність системи рівнянь. З іншого боку – для вищої точності моделювання нестационарних процесів необхідно більше цих вузлів, проте в міру, щоб, внаслідок малості змін невідомих витрат q , тисків P та температур T по просторовій координаті x з малим кроком $\Delta x = x_j - x_{j-1}$, де x_j – вузли розбиття ребра, $j = (\overline{3, n})$, не отримати систему із залежними рівняннями (з певною точністю). Тому треба знайти компроміс між швидкістю та стійкістю розв'язування систем рівнянь.

Як було зазначено вище, зменшення розмірності системи рівнянь – моделі системи, полягає в редагуванні розрахункової схеми, представленої у вигляді графа. Очевидним є той факт, що розрахунковий граф не повинен містити ребер нульової довжини чи діаметра, тому такі ребра ототожнюються з вершиною (сюди входять й відкриті крани, байпаси, клапани). Доцільно також розглядати послідовність ребер, які мають однаковий діаметр, як одне ребро. Тобто, якщо довільні суміжні ребра $e_j = (v_{i-1}, v_i)$, $e_{j+1} = (v_i, v_{i+1})$ мають однаковий діаметр $|D_{e_j} - D_{e_{j+1}}| < \varepsilon_D$, де ε_D – допустиме відхилення значень діаметра, то їх можна замінити одним ребром $e_j^* = (v_{i-1}, v_{i+1})$, вилучивши з графу вершину v_i , ребра e_j та e_{j+1} та присвоївши значення довжини нового ребра, яка дорівнює сумі двох об'єднаних $L_{e_j^*} = L_{e_j} + L_{e_{j+1}}$, а $D_{e_j^*} = D_{e_j}$. Певним параметром, який

спростить систему рівнянь, ϵL_{sh} – мінімальна довжина ребра у графі. Якщо довільне ребро ϵ коротшим, ніж L_{sh} , то ним нехтують (ребро ототожнюється з однією із вершин). Це зменшує кількість ребер, а отже, й кількість рівнянь. До вибору величини L_{sh} треба поставитись обережно, з огляду на те, щоб геометричний об'єм ребер редагованого графу незначно відрізнявся від оригінального графу ГТС, а також щоб суттєво не змінилась топологія (з огляду на це передбачена також функціональність присвоєння мітки ребру, яка не дозволить ототожнити його з вершиною, навіть попри те, що $L < L_{sh}$). Тому для кранів (усіх типів), які містять умову про зміну їх стану (закритий/відкритий), існує глобальний прапорець, який вказує на те, що ребра, інцидентні вершинам ребра – крана, ϵ автоматично ребрами які не підлягають ототожненню з відповідними вершинами.

Також треба зазначити, що певні розрахункові параметри вершини v (тиск чи надходження/відбір газу), яка вилучається з графу (коли довжина її ребра нульова або менша за L_{sh}), слід врахувати у вершині початку v_{noc} або кінця v_{kin} результуючого ребра e_R . В алгоритмі реалізоване правило вибору вершини з урахуванням відстані до відповідних вершин, тобто якщо $L(\overline{v, v_{noc}}) \leq L(\overline{v, v_{kin}})$, то зміни внесуться у вершину v_{noc} .

Опорними процедурами алгоритму стягування (спрощення вигляду графу технологічної схеми ГТС) ϵ дві процедури: об'єднання ребер [Процедура 1]; стягування ребер у вершину [Процедура 2]. Алгоритм стягування передбачає послідовне викликання процедур: [Процедура 1] – [Процедура 2] – [Процедура 1]. Розглянемо детальніше алгоритми цих процедур.

Зазначимо, що процедури використовують об'єкти ребер та вершин, які містять такі дані:

а) для ребер – список вершин вхідного графу, які об'єднало в себе ребро вихідного графу $ArVsh$, номери вершин, які стали початком цього ребра та кінцем – $Vsh1, Vsh2$ відповідно, списки вершин, які “влились” у ребро з початку та з кінця – $ArVsh1, ArVsh2$ відповідно, номер ребра e_R , якому належить це ребро у вихідному графі $iEsh$;

б) для вершин – номер ребра e_R ; прапорець відвідування вершини графу під час здійснення пошуку; список $ArVsh$ вершин, які в процесі стягування ϵ тотожними певній вершині; номер вершини, з якою отожднилася певна вершина $iVsh$.

Перед початком виконання кожної процедури здійснюється занулення необхідних масивів (списків) та значень для об'єктів ребер та вершин.

[Процедура 1]

1. Здійснюється прохід по всіх невідвіданих вершинах v_i , вершини-власники яких теж не відвідані. При цьому вершина не повинна: бути джерелом чи відбором газу, містити умову заданого тиску для розрахунку, містити крайові умови на тиск чи витрату газу, містити умову про заборону стягування її чи вершини-власника. Для кожної вершини v_i знаходять два суміжні ребра $e_1 = (v_1, v_i)$ і $e_2 = (v_i, v_2)$, якщо існують. Після цього здійснюється перевірка того, чи не ϵ вже відвіданими вершини v_1 та v_2 або їхні власники. Якщо виконається умова, що $|D_{e_1} - D_{e_2}| < \epsilon_D$, e_1 і e_2 ϵ трубою чи краном (довільного типу), то вершина v_i позначається як відвідана. Досумовуються довжини ребер L_{e_1} та L_{e_2} до змінної майбутньої довжини L_R результуючого ребра e_R , яке об'єднає кілька ребер. Після цього відбувається рух в одну та іншу сторону від вершини згідно з логікою пункту 2, тобто рух з вершини v_1 та v_2 .

2. Нехай здійснюється рух графом у напрямку $(\overrightarrow{v_i, v_1})$ (“*” – для руху у напрямку $(\overrightarrow{v_i, v_2})$ логіка аналогічна). Якщо вершина $v_1 (v_2)^*$ не була відвідана чи вершина-власник, не стоїть умова про заборону стягування її чи вершини-власника, не є джерелом чи відбором і знайдене наступне ребро $e_k = (v_1, v_1^1) (e_k = (v_2, v_2^1))^*$, то перевіряється, чи ребро e_k є ділянкою газопроводу і $|D_{e_1} - D_{e_k}| < \varepsilon_D$ або є краном (усіх видів) і не містить умови заборони стягування через те, що містить умову. Якщо всі перевірки виконано, то позначаються вершина $v_1 (v_2)^*$ та вершина-власниця (якщо існує) як відвідані, досумовується довжина ребра L_{e_k} до результуючої L_R , додається номер вершини $v_1 (v_2)^*$ в список $ArVsh1 (ArVsh2)^*$ ребра e_R , присвоюється початком $Vsh1$ (кінець $Vsh2$) ребра e_R вершині $v_1^1 (v_2^1)^*$. Повторюємо пункт 2, продовжуючи рух в напрямку $(\overrightarrow{v_1, v_1^1}) ((\overrightarrow{v_2, v_2^1}))^*$.
3. Вибираємо інший напрямок $(\overrightarrow{v_i, v_2})$ та повторюємо пункт 2. Отримаємо список $ArVsh2$ та $Vsh2$.
4. Об’єднуємо списки $ArVsh1$ та $ArVsh2$ у список $ArVsh$, вершині v_i та всім із цього списку присвоюємо $iEsh$ номер ребра e_R .
5. Всі зміни затверджуються за умови, що $L_R \geq L_{sh}$.

[Процедура 2]

1. Здійснюється прохід по всіх невідвіданих вершинах v_i , в яких не проставлено значення $iEsh$. Для кожної вершини v_i ставиться помітка про те, що вона відвідана і вноситься у стек.
2. Поки стек не порожній, то зі стека дістається остання вершина v_n (вилучається зі стека) і для неї знаходиться список суміжних вершин $v_j, j = \overline{1, k}$, де k – кількість суміжних вершин.
3. Здійснюється прохід по всіх v_j , які не були відвідані і в яких не проставлено значення $iEsh$. Розглядаємо відповідне ребро $e_j = (v_i, v_j)$. Якщо воно є краном (усіх типів), або не є: КС, краном (усіх типів), редуктором, пиловловлювачем, поглиначем чи холодильником і при тому $D_{e_j} < \varepsilon_D$ (нульовий діаметр), або $L_{e_j} < L_{sh}$ і його можна стягувати, то ребро e_j занулюється, у список $ArVsh$ вершини v_i додається вершина v_j , вершина v_j відмічається як відвідана, прописується вершина-власниця вершини v_j $iVsh = v_i$, якщо була помітка про те, що вершина не повинна стягнутись, то вона переписується вершині-власниці. Узгоджуються умови вершин v_i та v_j .
4. Вершина v_j долучається до стека й виконуються пункти 3-4.

Приклади роботи алгоритму

Приклад 1

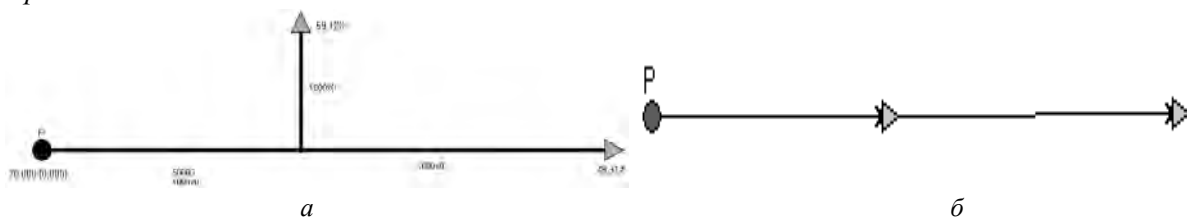


Рис. 1. а – граф-схема в оригіналі; б – граф-схема стягнута ($L_{sh} = 501$)

Приклад 2

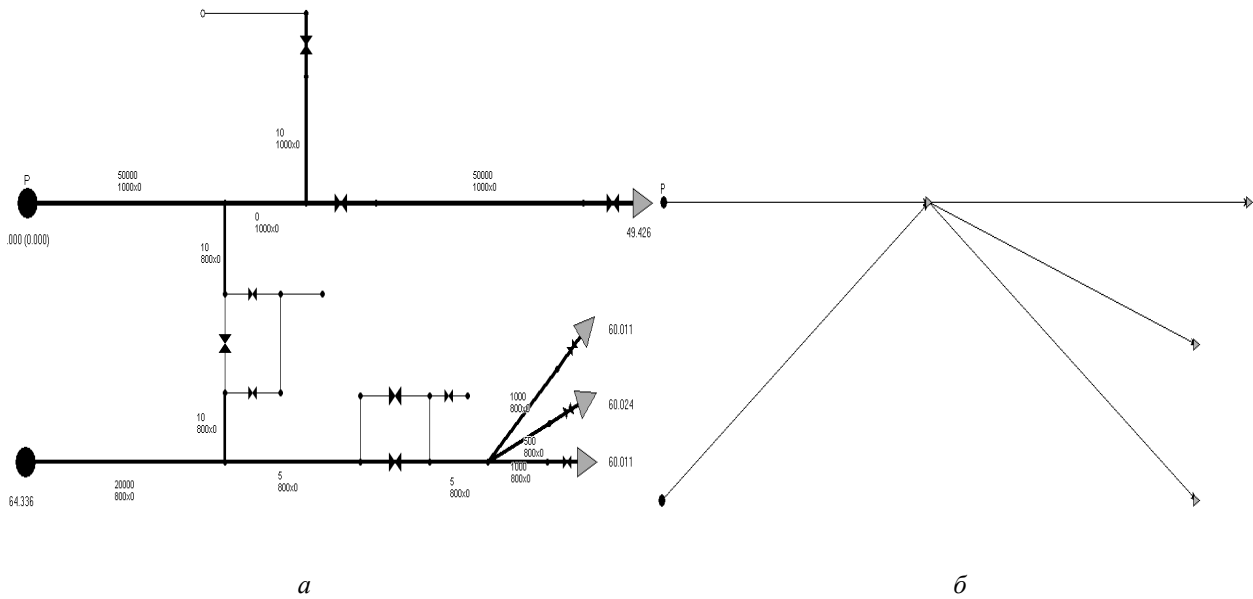


Рис. 2. а – граф-схема в оригіналі; б – граф-схема стягнута ($L_{sh} = 501$)

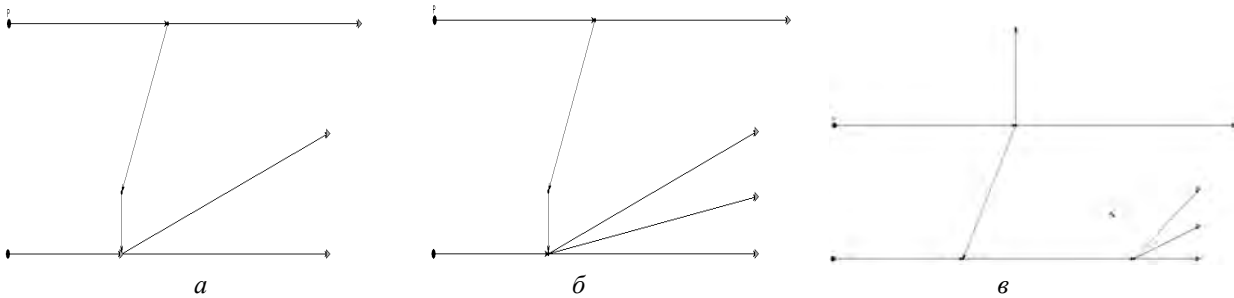


Рис. 3. а – граф-схема стягнута ($L_{sh} = 501$, перемичка з умовою і виставлена опція не стягувати сусідні труби до крана з умовою); б – граф-схема стягнута ($L_{sh} = 100$, перемичка з умовою і виставлена опція не стягувати сусідні труби до крана з умовою); в – граф-схема стягнута ($L_{sh} = 1$)

Висновки

Запропоновані алгоритми пройшли апробацію на великій кількості реальних прикладів. Аналіз результатів підтвердив ефективність роботи розроблених алгоритмів. Результати розрахунків відрізнялися від заміряних на величину, достатню для проведення моделювання газодинамічних процесів, які проходять у газотранспортній системі.

1. Панкратов В.С. Автоматизированная система диспетчерского управления ГТС / В.С. Панкратов, А.С. Верблю. – М.: Изд-во ООО "ИРЦ Газпром", 2002. – 98 с. 2. Сарданашвили С.А. Расчетные методы и алгоритмы / С. А. Сарданашвили. – М.: Нефть и газ, 2005. – 577 с. 3. Селезнев В. Е. Методы и технологии численного моделирования газопроводных систем / В. Е. Селезнев, В. В. Алешин, Г. С. Клишин. – М.: Эдиториал УРСС, 2002. – 448 с. 4. Бобровский С. А. Трубопроводный транспорт газа / С. А. Бобровский, С. Г. Щербаков, Е. И. Яковлев. – М.: Наука, 1976. – 475 с. 5. Сінчук Ю. Моделювання роботи кранів на магістральних газопроводах / Ю. Сінчук, Н. Притула, М. Притула // Вісник Національного університету "Львівська політехніка": Комп'ютерні науки та інформаційні технології. – Львів. – 2010. – № 663. – С. 216–222. 6. Сінчук Ю., Притула Н., Притула М. Моделювання нестационарних режимів газових мереж // Вісник Національного університету "Львівська політехніка". Комп'ютерні науки та інформаційні технології. – Львів. – 2010. – № 663. – С. 128–132.