**УДК 519.876.3**

**Б. Каменський**
Львівський національний університет імені Івана Франка,
механіко-математичний факультет

# АВТОМАТИЗАЦІЯ ПОБУДОВИ ДИНАМІЧНИХ МОДЕЛЕЙ І ОПТИМІЗАЦІЯ ВИРОБНИЧИХ СИСТЕМ



**Розглянуто способи представлення абстрактних технологічних процесів і мережевих графіків у вигляді зважених графів. Подано чітку аксіоматику понять абстрактного технологічного процесу та способів його представлення. Показано переваги вершинного представлення. Побудовано алгоритм знаходження властивостей технологічного процесу та його операцій та доведено коректність його побудови. Розглянуто приклади різних представлень.**

**Some ways of representation of abstract workflow and project networks as weighted graphs are considered. Exact axiomatic of concepts of abstract workflow and ways of its representation is given. The advantages of vertex representation are demonstrated. The algorithm of finding workflow's and operation's properties is proposed and its correctness is shown. Some examples of different ways of representation are considered.**

## Вступ

В основу ефективних систем оперативного керування складними виробничими системами покладено моделюючі системи. Якщо виробничі системи складаються з великої кількості об'єктів, котрі тісно між собою взаємодіють згідно з єдиним технологічним процесом, то виникають ще задачі автоматизації процесу побудови їх моделей. До таких складних виробничих систем належить залізниця. Всі станції, її парки спеціалізуються на наборах операцій, об'єднаних в спеціальні групи, які називаються технологічним процесом. Всі станції мають свої особливості: колійний розвиток, технічне оснащення, кадрове забезпечення тощо. Основна задача такої виробничої системи – оптимальна організація вагоно- і поїздопотоків. Вона пов'язана із розв'язанням великої кількості динамічних задач. Ефективність їх розв'язання залежить від системи моделювання тих процесів, які проходять у таких системах.

Роботу станції можна подати у вигляді набору зв'язних технологічних процесів. Основними елементами технологічних процесів є технологічні операції, які описуються набором параметрів. Зручно технологічні процеси представляти у вигляді певних зважених графів. При цьому виникають певні питання інтерпретації виробничих процесів в теоретико-графових термінах. Цим та іншим питанням присвячена робота.

Описані теоретико-графові об'єкти виникають також в задачах керівництва проектами та багатьох інших задачах. Діаграма PERT (PERT chart, абревіатура PERT розшифровується як program evaluation and review technique – система планування та керівництва розробками), мережевий графік – це поняття, що відповідають поняттю технологічного процесу. Ширше застосування дає змогу використовувати результати роботи також для постановки і розв'язування багатьох інших задач.

## Поняття абстрактного технологічного процесу

***Відношенням*** на множині $A$ назвемо підмножину декартового добутку $B \subset A \times A$. Відношення $B$ на множині $A$ буде ***антитранзитивним***, якщо воно задовольняє таку умову:

$$\forall x, y, z \in A : (x, y) \in B \wedge (y, z) \in B \Rightarrow (x, z) \notin B .$$

Послідовність елементів множини $A$ утворює м***аршрут*** $M$ ***на множині*** $A$ ***стосовно відношення*** $B$, якщо кожні сусідні елементи цієї послідовності як впорядкована пара належать відношенню $B$: $M = (x_0, \ldots, x_n)$ – маршрут, якщо

$$\forall i \in \overline{0,n} \;\; x_i \in A \;\wedge\; \forall i \in \overline{1,n} \;\; (x_{i-1}, x_i) \in B .$$

Якщо $x_0 = x_n$, то маршрут $M$ назвемо ***замкненим***. ***Циклом*** $C$ ***на множині*** $A$ ***стосовно відношення*** $B$ назвемо такий замкнений маршрут, в якого всі елементи різні (крім першого і останнього, які збігаються):

$$C = (x_0, \ldots, x_{n-1}, x_0) \text{ – цикл, якщо } \forall i, j \in \overline{0, n-1} : i \neq j \Rightarrow x_i \neq x_j .$$

Відношення $B$ на множині $A$ буде ***ациклічним***, якщо на множині $A$ стосовно відношення $B$ не існує циклів. Якщо відношення $B$ – ациклічне, то воно антирефлексивне: $\forall x \in A : (x, x) \notin B$ (умова відсутності петель) і асиметричне $\forall x, y \in A : (x, y) \in B \Rightarrow (y, x) \notin B$ (умова відсутності зворотних ребер). Відношення $B$ на множині $A$ є ***відношенням безпосереднього передування*** або ***технологічним відношенням,*** якщо воно антитранзитивне і ациклічне. ***Вага елемента множини*** $A$ – це функція, що ставить у відповідність кожному елементу множини $A$ деяке невід’ємне дійсне число.

Впорядковану трійку $A = (O, R, l)$ називатимемо а***бстрактним технологічним процесом***, або абстрактним мережевим графіком. Тут $O$ – множина операцій; $R \subset O \times O$ – ***технологічне відношення*** на множині операцій; $l : O \to \mathbf{R}_+$ – ***вага***, або ***функція тривалості*** операцій.

Будемо говорити, що операція $x \in O$ ***безпосередньо передує*** операції $y \in O$, або що ***виконання операції*** $x$ ***безпосередньо передує виконанню операції*** $y$, якщо $(x, y) \in R$.

Якщо $x$ ***безпосередньо передує*** операції $y$, то операцію $x \in O$ назвемо ***безпосереднім попередником*** операції $y \in O$, а операцію $y \in O$ – ***безпосереднім послідовником*** операції $x \in O$.

Аксіоматичному поняттю абстрактного технологічного процесу відповідають як інтуїтивні уявлення про технологічний процес, так і інтуїтивні уявлення про мережевий графік, адже обидва ці поняття передбачають задання: переліку всіх операцій процесу чи проекту (множина $O$); часу, необхідного для виконання кожної операції (функція $l : O \to \mathbf{R}_+$); переліків операцій, які безпосередньо передують кожній операції (відношення безпосереднього передування $R \subset O \times O$).

### Реберний і вершинний способи представлення процесу

Введемо поняття джерела і стоку для орієнтованого ациклічного графу $G = (V, E)$. Тут і надалі ми будемо розглядати лише орієнтовані ациклічні графи, тому під словом “граф” будемо розуміти “орієнтований ациклічний граф”.

**Аксіома (аксіома джерела).**

$$\exists! S \in V : \; \forall e = (v_1, v_2) \in E \;\; v_2 \neq S$$

Якщо для даного графу $G = (V, E)$ виконується аксіома джерела, то вершину $S$ з цієї аксіоми називаємо ***джерелом***.

**Аксіома (аксіома стоку).**

$$\exists! T \in V : \; \forall e = (v_1, v_2) \in E \;\; v_1 \neq T .$$

Якщо для даного графу $G = (V, E)$ виконується аксіома стоку, то вершину $T$ з цієї аксіоми називаємо ***стоком***.

Введемо поняття вагової функції для ребер і вершин відповідно до двох різних представлень, поданих нижче.

***Вагова функція ребер***, або ж ***вага ребра*** – це функція, що ставить у відповідність кожному ребру графу $G = (V, E)$ деяке невід’ємне дійсне число.

***Вагова функція вершин***, або ж ***вага вершини*** – це функція, що ставить у відповідність кожній вершині графу $G = (V, E)$ деяке невід’ємне дійсне число.

Існують два принципово різні способи представлення технологічного процесу у вигляді ациклічного орієнтованого графу.

• ***Реберний спосіб представлення.*** Операції відповідають дугам (орієнтованим ребрам), а зв'язки між операціями відповідають вершинам.

• ***Вершинний спосіб представлення***. Операції відповідають вершинами, а зв'язки, що вказують на порядок між операціями, відповідають ребрам.

У роботі реалізовано вершинний спосіб представлення. У цьому випадку виникає необхідність введення двох фіктивних вершин-операцій: ***Старт*** – початок всіх операцій, і ***Фініш*** – завершення всіх операцій. У випадку реберного способу представлення виникає необхідність введення фіктивних і паралельних дуг.

***Реберним способом представлення абстрактного технологічного процесу*** $A=(O,R,l)$ як зваженого орієнтованого ациклічного графу, який задовольняє аксіоми джерела і стоку, будемо називати таку впорядковану шістку $G_1=(V_1,E_1,S_1,T_1,F_1,w_1)$, де:

• $V_1$ – множина вершин цього графу;

• $E_1$ – множина ребер цього графу;

• $S_1$ – єдине джерело цього графу;

• $T_1$ – єдиний стік цього графу;

• $F_1 \subset E_1$ – підмножина фіктивних дуг;

• $w_1 : E_1 \to \mathbf{R}_+$ – вагова функція ребер, яка задовольняє ***умову відсутності ваги фіктивних дуг*** $\forall f \in F_1 \;\; w_1(f)=0$,

якщо виконуються такі умови узгодженості:

• $\exists f : O \mapsto O_1 = E_1 \setminus F_1$ – бієкція;

• $\forall x,y \in O : (x,y)\in R \;\; \exists p=\left\{e_i=\left(a_{i-1},a_i\right)\right\}_{i\in\overline{1,n}} \in E_1^n, n\ge 2 : e_1=f(x)\wedge e_n=f(y)$ – якщо операція $x$ безпосередньо передує операції $y$, то існує маршрут, що починається з ребра $f(x)$ і закінчується ребром $f(y)$;

• $\forall x \in O \;\; \exists p=\left\{e_i=\left(a_{i-1},a_i\right)\right\}_{i\in\overline{1,n}} \in E_1^n, \exists k\in\overline{1,n} : e_k=f(x)\wedge a_0=S_1\wedge a_n=T_1$ – для кожної операції $x$ існує маршрут, що починається із джерела, закінчується у стоці і містить ребро $f(x)$;

• $\forall x \in O \;\; l(x)=w(f(x))$ : вага ребер – це тривалість операцій.

Реберний спосіб представлення $G_1=(V_1,E_1,S_1,T_1,F_1,w_1)$ використовується в поняттях діаграми PERT і мережевого графіка. В теорії мережевих графіків елементи множини $V_1$ називаються ***подіями***, джерело $S_1$ називається ***початковою подією***, стік $T_1$ називається ***кінцевою подією***. Умова ациклічності відношення $R$ накладає умову відсутності контурів у графі $G_1$.

***Вершинним способом представлення абстрактного технологічного процесу*** $A=(O,R,l)$ як зваженого орієнтованого ациклічного графу, який задовольняє аксіоми джерела і стоку, будемо називати таку впорядковану п'ятірку $G_2=(V_2,E_2,S_2,T_2,w_2)$, де:

• $V_2$ – множина вершин цього графу;

• $E_2$ – множина ребер цього графу;

• $S_2$ – єдине джерело цього графу, яке ми будемо називати ***стартом***;

• $T_2$ – єдиний стік цього графу, який ми будемо називати ***фінішом***;

• $w_2 : V_2 \to \mathbf{R}_+$ – вагова функція вершин, яка задовольняє ***умову відсутності ваги фіктивних вершин*** $w_2(S_2)=0 \wedge w_2(T_2)=0$,

якщо виконуються такі умови узгодженості:

• $\exists f : O \mapsto V_2$ – бієкція;

• $\forall x, y \in O : (x, y) \in R \ \ \left( f(x), f(y) \right) \in E_2$ – якщо операція $x$ безпосередньо передує операції $y$, то існує ребро, що починається у вершині $f(x)$ і закінчується у вершині $f(y)$;

• $\forall x \in O \ \ \exists p = \left\{ \left( a_{i-1}, a_i \right) \right\}_{i \in \overline{1,n}} \in E_1^n, \exists k \in \overline{1, n-1} : a_k = f(x) \wedge a_0 = S_1 \wedge a_n = T_1$ – для кожної операції $x$ існує маршрут, що починається зі старту, закінчується у фініші і проходить через вершину $f(x)$;

• $\forall o \in O \ \ l(o) = w(f(o))$: вага вершин – це тривалість операцій.

Умова ациклічності відношення $R$ накладає умову відсутності циклів у графі $G_2$.

Під час роботи були доведені такі твердження.

**Твердження 3.1.** (Побудова реберного представлення для абстрактного технологічного процесу)

Для кожного абстрактного технологічного процесу $A = (O, R, l)$ існує його реберне представлення $G_1 = (V_1, E_1, S_1, T_1, F_1, w_1)$.

**Твердження 3.2.** (Побудова вершинного представлення для абстрактного технологічного процесу)

Для кожного абстрактного технологічного процесу $A = (O, R, l)$ існує його вершинне представлення $G_2 = (V_2, E_2, S_2, T_2, w_2)$.

Позначимо через $A$ множину всіх можливих реберних представлень, а через множину $B$ – множину всіх можливих вершинних представлень.

**Твердження 3.3.** $\exists F : A \to B$ – сюр'єкція. Тобто, існує сюр'єктивне і неін'єктивне відображення з множини всіх можливих реберних представлень на множину всіх можливих вершинних представлень.

**Переваги вершинного способу представлення**

• Зведення кількості фіктивних операцій до мінімуму: старт, фініш.

• Єдиність представлення;

• Відсутність паралельних дуг.

### Технологічні процеси як приклад вершинного представлення

Обробку кожного вагона на станції можна розбити на велику кількість різних технологічних операцій. Деякі з цих операцій можуть виконуватися одночасно, інші – тільки послідовно: одна операція після закінчення іншої.

Для задання абстрактного технологічного процесу $A = (O, R, l)$ потрібно визначити:

1. Перелік всіх операцій проекту (множина $O$).
2. Час, необхідний для виконання кожної операції (вага $l$).
3. Переліки операцій, які безпосередньо передують кожній операції (відношення $R$).

***Технологічний процес*** – це таке вершинне представлення деякого абстрактного технологічного процесу $A = (O, R, l)$ як зваженого орієнтованого ациклічного графу $G = (V, E, S, T, w)$, який реалізований програмно за допомогою списку суміжних вершин, де:

• $V$ – множина вершин, яка однозначно відповідає множині ***технологічних операцій*** $O$, тобто, існує бієкція $\exists f : O \mapsto V$;

• $E$ – множина ребер відповідає ***технологічному відношенню*** $R$ так, що якщо операція $x$ ***безпосередньо передує*** операції $y$, то існує ребро, що починається з вершини $f(x)$, і закінчується у вершині $f(y)$;

• $S$ – єдине джерело, яке ми будемо називати ***стартом***;

• $T$ – єдиний стік, який ми будемо називати ***фінішом***;

• $w : V \to \mathbf{R}_+$ – вагова функція вершин, яка дорівнює нулю для старту і фінішу і яка дорівнює тривалості операції для всіх інших вершин;

і для якого виконується умова:

• $\forall x \in O \ \exists p = \left\{ \left( a_{i-1}, a_i \right) \right\}_{i \in \overline{1,n}} \in E_1^n, \exists k \in \overline{1, n-1} : a_k = f(x) \wedge a_0 = S_1 \wedge a_n = T_1$ – для кожної операції $x$ існує маршрут, що починається зі старту, закінчується у фініші, і проходить через вершину $f(x)$.

## Властивості технологічного процесу і його операцій

Важливою характеристикою технологічного процесу $G=(V,E,S,T,w)$ є його тривалість, яка визначається вагою критичного шляху. ***Вага*** шляху $p=\langle v_0,v_1,\ldots,v_k\rangle$ дорівнює сумарній вазі всіх початків дуг, що входять у шлях:

$$w(p)=\sum_{i=0}^{k-1}w(v_i)\,,$$

а ***вага критичного шляху*** $\delta(u,v)$ з вершини $u$ до вершини $v$ визначається співвідношенням:

$$\delta(u,v)=\sup_{p}\left\{w(p)\mid p=\langle v_0,v_1,\ldots,v_k\rangle : v_0=u, v_k=v\right\}.$$

Тут супремум дорівнює максимуму серед ваг існуючих шляхів, якщо такі існують, або мінус безмежності, якщо таких шляхів не існує.

Шлях $p=\langle v_0,v_1,\ldots,v_k\rangle$ – ***критичний***, якщо його вага дорівнює вазі критичного шляху:

$$w(p)=\delta(v_0,v_k).$$

***Тривалість*** $Dur(G)$ технологічного процесу характеризується вагою критичного шляху зі старту до фінішу:

$$Dur(G)=\delta(S,T).$$

Введене означення критичного шляху є коректним для ***ациклічних*** графів. Важливим є те, що критичний шлях має властивість оптимальної підструктури, що означає можливість застосування скупих алгоритмів чи динамічного програмування для його знаходження.

Справедлива така лема.

**Лема 5.1** (Властивість оптимальної підструктури: часткові шляхи критичного шляху є критичними шляхами.) Нехай $p=\langle v_0,v_1,\ldots,v_k\rangle$ – критичний шлях з вершини $v_0$ до вершини $v_k$ у заданому зваженому орієнтованому ациклічному графі $G=(V,E)$ з ваговою функцією $w:V\to\mathbf{R}_+$, а $p_{ij}=\langle v_i,v_{i+1},\ldots,v_j\rangle$ – частковий шлях шляху $p$, що проходить з вершини $v_i$ до вершини $v_j$ для довільних $i$ і $j$, що задовільняють нерівності $0\le i\le j\le k$. Тоді $p_{ij}$ – критичний шлях з вершини $v_i$ до вершини $v_j$.

Кожна операція $i\in V$ характеризується ***найбільш раннім терміном початку операції*** ($E_i$), тобто найбільш раннім з можливих термінів настання операції, а також – ***найпізнішим терміном завершення операції*** ($L_i$), що ще допускає своєчасного закінчення всього проекту.

***Найбільш ранній термін початку операції*** $E_i$ визначається вагою критичного шляху від старту до операції $i$:

$$E_i=\delta(S,i),$$

а ***найпізніший термін завершення операції*** $L_i$ – сумою різниці тривалості технологічного процесу та ваги критичного шляху від операції $i$ до фінішу і ваги операції $i$:

$$L_i=Dur(G)-\delta(i,T)+w(i).$$

Розглянемо деяку операцію $v\in V$. Виникає питання про максимальну кількість часу, яку можна виділити для її виконання без затримки своєчасного закінчення виконання всього технологічного процесу. Операція $v\in V$ може початися не раніше терміну $E_v$ і повинна закінчитися не пізніше $L_v$. Отже, без затримки завершення процесу на виконання операції $v\in V$ можна виділити не більше, ніж $L_v-E_v$ одиниць часу. Отже, при виконанні цієї операції можна допустити максимальну затримку $L_v-E_v-w(v)\ge 0$.

Величина $L_v-E_v-w(v)$ – це ***повний резерв часу операції*** $v\in V$.

Очевидно, що затримка виконання операції, повний резерв часу якої дорівнює нулю, приведе до такої самої за часом затримки виконання всього проекту.

Операція називається ***критичною***, якщо будь-яка затримка в її виконанні призводить до затримки завершення всього технологічного процесу. Іншими словами, критична операція – це операція, повний резерв часу якої дорівнює нулеві: $L_v - E_v - w(v) = 0$ . Можна вважати операцію ***критичною***, якщо вона лежить на критичному шляху від старту до фінішу. Під час роботи було доведено, що ці поняття є еквівалентними.

Знання всіх критичних операцій необхідне для запобігання затримці їхнього виконання, що затримує закінчення технологічного процесу. Для некритичних операцій можна допустити затримку, яка не перевищує повного резерву, не порушуючи своєчасності закінчення технологічного процесу.

Дерево критичних шляхів з коренем у вершині $S$ – це орієнтований підграф $G' = (V', E')$ , у якому множини $V' \subseteq V$ і $E' \subseteq E$ визначаються такими умовами:

1. $V'$ – множина вершин, досяжних із джерела $S$ графу $G$ ;

2. Граф $G'$ утворює кореневе дерево з коренем у вершині $S$ ;

3. Для всіх $v \in V$ однозначно визначений простий шлях з вершини $S$ до вершини $v$ у графі $G'$ збігається з критичним шляхом з вершини $S$ до вершини $v$ у графі $G$ .

Масив $CP$ буде реалізовувати дерево критичних шляхів в тому сенсі, що для кожної вершини $v \in V$ $v$ -та комірка масиву $CP$ : $CP_v$ містить або індекс іншої вершини, або значення $-1$, яке означає відсутність попередника. В алгоритмі значення $CP_v$ будуть присвоюватися так, що ланцюжок попередників, який починається у вершині $v$ , дозволятиме прослідкувати шлях, обернений до критичного шляху з вершини $u$ до вершини $v$ . Отже, знаючи масив $CP$ , можна буде побудувати критичний шлях з вершини $u$ до вершини $v$ для всіх вершин $\forall u, v \in V$ або вказати, що такого шляху не існує.

Наведемо приклад технологічного процесу

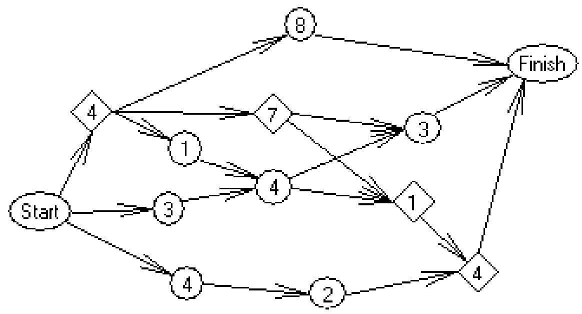


*Рис. 1. Технологічний процес з виділеним критичним шляхом*

На рис. 1 зображено технологічний процес, на якому критичні операції позначені ромбоподібним контейнером.

**Постановка задачі і алгоритм розв'язування**

Для заданого технологічного процесу $G$ потрібно знайти такі його параметри:

• тривалість технологічного процесу $Dur(G)$ ;

• масив дерева критичних шляхів $CP$ ;

• масив найбільш ранніх термінів початку операцій $E$.

• масив найпізніших термінів завершення операцій $L$.

Алгоритм знаходження перерахованих параметрів технологічного процесу складається із шести кроків, перерахованих нижче. Ідеї алгоритму запозичені з алгоритму відшукання найкоротших шляхів з однієї вершини в орієнтованих ациклічних графах

(Dag_Shortest_Paths(G, w, s)) [1, с. 677], і з алгоритмів розрахунку найбільш ранніх термінів настання подій і розрахунку найбільш пізніх термінів подій [2, с. 295–298]. Цей алгоритм містить такі нововведення: по-перше, у нашій моделі ми присвоюємо вагу не *ребрам*, а *вершинам*; по-друге, ми шукаємо критичний, а не найкоротший шлях; і по-третє, ця реалізація містить поєднання різних ідей в одному алгоритмі.

**Крок 1.** Топологічне сортування вершин технологічного процесу.

Викликати пошук в глибину DFS(*G*) для технологічного процесу $G$ з аргументом $S$ з модифікацією: після завершення роботи над вершиною записати її в останнє вільне місце в масиві пересортування.

**Крок 2.** Процедура $\mathrm{Init}(G,S)$. Ініціалізація масивів *E* та *CP*.

$\forall i \in I : E_i := -\infty$,

$\forall i \in I : CP_i := -1$,

$E_S := 0$.

**Крок 3.** Для кожної вершини *u* у порядку топологічного сортування і для кожної вершини *v* зі списку суміжних вершин вершини *u* викликати процедуру *Послаблення* (*u*, *v*):

Якщо $E_u + Dur(u) > E_v$, то $E_v := E_u + Dur(u) \wedge CP_v := u$.

**Крок 4.** Присвоїти значенню тривалості процесу найбільш ранній термін настання фінішу.

$Dur(G) := E_T$

**Крок 5.** Процедура $\mathrm{Init}\,2(G,T)$. Ініціалізація масиву *L*.

$\forall i \in I : L_i := +\infty$

$L_T := Dur(G)$

**Крок 6.** Для кожної вершини *u* у порядку, зворотньому до топологічного сортування, і для кожної вершини *v* зі списку суміжних вершин вершини *u* викликати процедуру *Послаблення2* (*u*, *v*).

Якщо $L_v - w(v) < L_u$, то $L_u := L_v - w(v)$.

Справедлива теорема.

**Теорема 7.1**. Якщо зважений орієнтований граф $G = (V, E)$ містить джерело $S$ і не містить циклів, то по завершенні алгоритму для усіх вершин $v \in V$ виконується рівність $E_v = \delta(S,v)$ і підграф передування $G_\pi$, реалізований масивом *CP*, являє собою дерево критичних шляхів.

Доведення теореми ґрунтується на доведених лемах (7.1–7.7), які сформульовані нижче.

**Лема 7.1** (Обернена нерівність трикутника). Нехай $G = (V, E)$ – зважений орієнтований ациклічний граф з ваговою функцією $w : V \to \mathbf{R}_+$ і джерелом $S$. Тоді для всіх ребер $(u,v) \in E$ виконується нерівність:

$$\delta(S,v) \geq \delta(S,u) + w(u).$$

**Лема 7.2** (Властивість верхньої границі). Нехай $G = (V, E)$ – зважений орієнтований ациклічний граф з ваговою функцією $w : V \to \mathbf{R}_+$ і джерелом $S$, ініціалізований процедурою $\mathrm{Init}(G,S)$. Тоді для всіх вершин $v \in V$ виконується нерівність $E_v \leq \delta(S,v)$, яка не змінюється (є інваріантом) впродовж всіх етапів послаблення графу *G*. Більше того, як тільки атрибут $E_v$ досягає своєї верхньої границі $\delta(S,v)$, він надалі не змінюється.

**Наслідок 7.1** (Властивість відсутності шляху). Нехай $G = (V, E)$ – зважений орієнтований ациклічний граф з ваговою функцією $w : V \to \mathbf{R}_+$ і джерелом $S$. Припустімо, що в графі $G$ відсутні шляхи, що з'єднують джерело $S \in V$ з даною вершиною $v \in V$. Тоді після ініціалізації графа процедурою $\mathrm{Init}(G,S)$ маємо $E_v = \delta(S,v) = -\infty$, і ця рівність зберігається як інваріант у ході виконання довільної послідовності кроків послаблення графа $G$.

**Лема 7.3** Нехай $G=(V,E)$ – зважений орієнтований ациклічний граф з ваговою функцією $w:V\to\mathbf{R}_+$, і нехай $(u,v)\in E$. Тоді безпосередньо після виконання процедури *Послаблення* (*u*, *v*) виконується нерівність $E_v \ge E_u + w(u)$.

**Лема 7.4** (Властивість збіжності). Нехай $G=(V,E)$ – зважений орієнтований ациклічний граф з ваговою функцією $w:V\to\mathbf{R}_+$ і джерелом $S$, ініціалізований процедурою $\mathrm{Init}(G,S)$. Нехай також $p=\langle v_0,v_1,\ldots,v_k\rangle : v_0=S\wedge v_{k-1}=u\wedge v_k=v$ – критичний шлях для деяких його вершин $u,v\in V$. Припустимо, після ініціалізації графу $G$ виконана послідовність етапів послаблення, що включає виклик процедури *Послаблення* (*u*, *v*). Якщо в деякий момент часу до виклику виконується рівність $E_u=\delta(S,u)$, то в будь-який момент часу після виклику справедлива рівність $E_v=\delta(S,v)$.

**Лема 7.5** (Властивість послаблення шляху). Нехай $G=(V,E)$ – зважений орієнтований ациклічний граф з ваговою функцією $w:V\to\mathbf{R}_+$ і джерелом $S$, ініціалізований процедурою $\mathrm{Init}(G,S)$. Розглянемо довільний критичний шлях $p=\langle v_0,v_1,\ldots,v_k\rangle : v_0=S\wedge v_k=v$ із джерела $S=v_0$ у вершину $v=v_k$. Припустимо, після ініціалізації графу $G$ виконана послідовність етапів послаблення з параметрами $(v_0,v_1),(v_1,v_2),\ldots,(v_{k-1},v_k)$ у зазначеному порядку, то після цих послаблень, а також в будь-який момент часу опісля виконується рівність $E_v\le\delta(S,v)$. Ця властивість справедлива незалежно від того, чи виконуються послаблення з іншими параметрами, включаючи послаблення, що чергуються з даними послабленнями.

**Лема 7.6**. Нехай $G=(V,E)$ – зважений орієнтований ациклічний граф з ваговою функцією $w:V\to\mathbf{R}_+$ і джерелом $S$, ініціалізований процедурою $\mathrm{Init}(G,S)$. Тоді після ініціалізації графа процедурою $\mathrm{Init}(G,S)$ підграф передування $G_\pi$ утворює кореневе дерево з коренем $S$, а будь-яка послідовність кроків послаблення ребер графа $G$ підтримує цю властивість як інваріант.

**Лема 7.7** (Властивість підграфу передування). Нехай $G=(V,E)$ – зважений орієнтований ациклічний граф з ваговою функцією $w:V\to\mathbf{R}_+$ і джерелом $S$, ініціалізований процедурою $\mathrm{Init}(G,S)$. Виконаємо довільну послідовність кроків послаблення графу $G$, у результаті якої для усіх вершин $v\in V$ виконується рівність $E_v=\delta(S,v)$. Тоді підграф передування $G_\pi$ – дерево критичних шляхів з коренем $S$.

### Система автоматизації побудови моделі виробничого об'єкта

В основу системи автоматизації побудови моделей виробничих об'єктів покладено розроблену базу типових технологічних процесів та система редагування технологічних процесів з графічною підтримкою. Це дає змогу швидко формувати технологічні процеси конкретних станцій і пов'язувати їх з існуючою оперативною базою ресурсного забезпеченням (база технічного, технологічного, кадрового, інформаційного і т.д. забезпечення). В такій системі розв'язується проблема ефективної підтримки бази технологічних процесів в актуальному стані за рахунок їхнього зв'язку з єдиною нормативною базою. На основі відомого для кожної станції виробничого процесу формується способом синтезу технологічних процесів динамічна модель її роботи. Маючи систему прогнозування поїздопотоків можна прогнозувати вихідні вагоно- і поїздопотоки для довільної станції (виробничого об'єкта) (рис. 2).

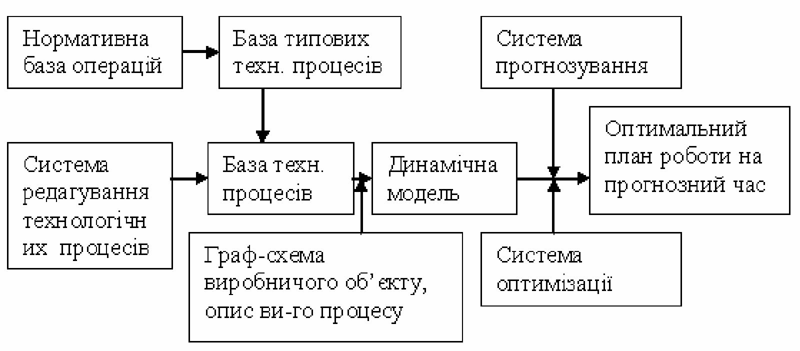


*Рис. 2. Система автоматизації процесу побудови моделі об'єкта та моделювання його виробничого процесу*

**Висновки**

Запропоноване і обгрунтоване в роботі представлення технологічного процесу у вигляді теоретико-графових об'єктів забезпечує однозначну інтерпретацію довільного абстрактного технологічного процесу і тому його без додаткових застережень можна використати для побудови довільного існуючого виробничого процесу.

Результати роботи дали можливість побудувати динамічну модель роботи станції та сформулювати задачі на максимальну переробну і пропускну її спроможність.

*1. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы: построение и анализ, 2-е изд.: Пер. с англ. – М.: Издательский дом "Вильямс", 2005. – 1296 с.: ил. – Парал. тит. англ. ISBN 5-8459-0857-4 (рус.) 2. Майника Э. Алгоритмы оптимизации на сетях и графах. – М.: Мир, 1981. – 324 с.*

**УДК 681.84.087.4**

**А. Ковальчук**
Національний університет "Львівська політехніка",
кафедра автоматизованих систем управління

# ПРО ДЕЯКІ НЕСКІНЧЕННІ МНОЖИНИ ПРОСТИХ ЧИСЕЛ ДЛЯ ВИКОРИСТАННЯ В СИСТЕМІ RSA



**Для квадратного полінома з цілими коефіцієнтами і від натуральної змінної розв'язана задача про потужність підмножини простих значень такого полінома. На підставі отриманого результату обгрунтовується можливість побудови алгоритмів вибору простих чисел для криптографічних систем з відкритими ключами.**

**For the quadratic polinomial with integer coefficients and at the natural variable the solving of the problem of the infinity sets of the prime values of this polynomial is presented. With a foundation on the recived result the possibility of the construction of the prime numbers for the kryptographic systems with open keys are based.**

**Вступ**

При побудові криптоалгоритмів для генерування системи ключів в системі RSA необхідно випадково вибирати прості числа для генерування відкритого ключа $N = p{\cdot}q$, від випадковості