

Optimizavimas ekonomikoje

K. Sydsæter, P. Hammond, A. Seierstad, A. Strøm

Further Mathematics for Economic Analysis, FT Prentice Hall, 2008.

M. D. Intriligator

Mathematical Optimization and Economic Theory, 2002.

V. Čiočys, R. Jasilionis

Matematinis programavimas, 1990.

N. Batarlienė, M. Mazūra

Tiesinio programavimo modeliai transporte, 2006.

A. K. Dixit

Optimization in Economic Theory, 1990.

M. Luptácik

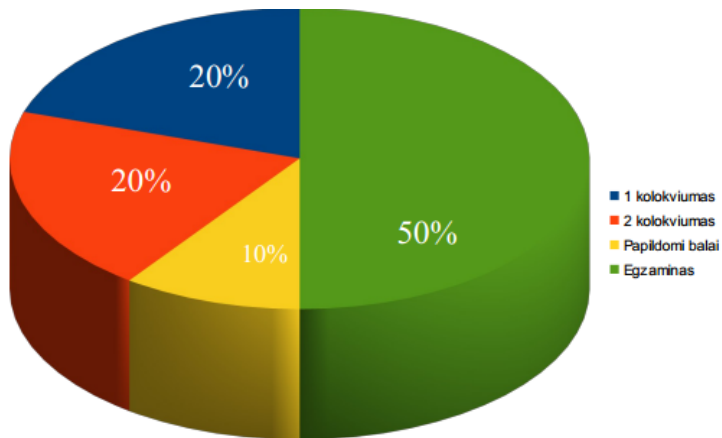
Mathematical Optimization and Economic Analysis, 2009.

J. Ekeland, R. Témam

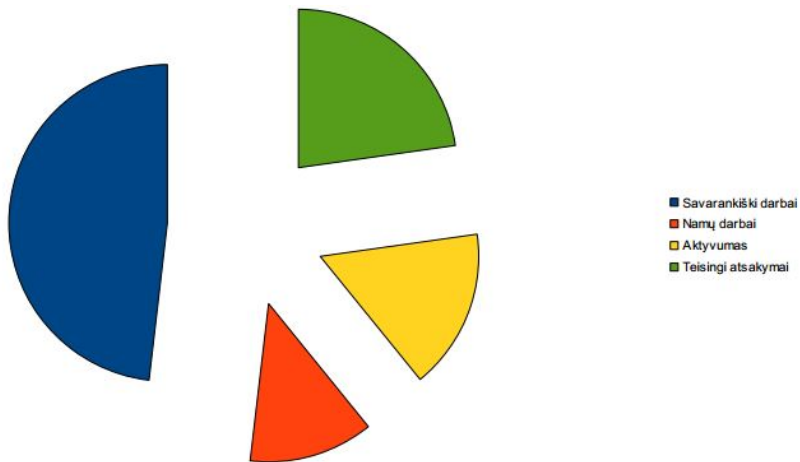
Convex Analysis and Variational Problems, 1999.

- Transporto uždaviniai
- Iškilusis programavimas. Ekonominė interpretacija.
- Netiesinis programavimas. Kuno-Takero sąlygos. Ekonominė interpretacija.
- Dinaminis programavimas.
- Stochastinis programavimas.
- Tinklinis planavimas.

Vertinimo tvarka (1)

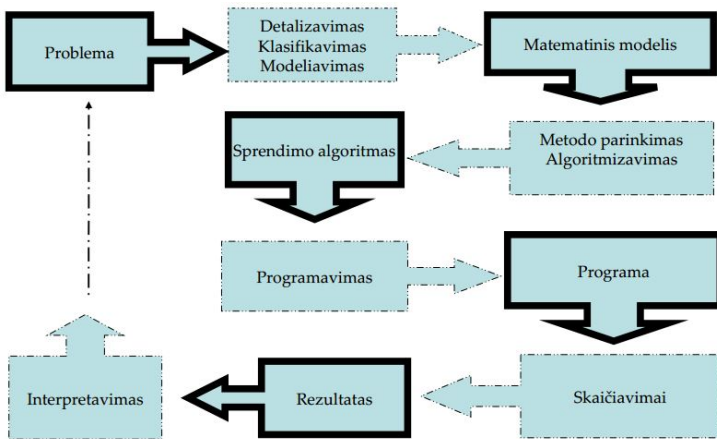


Vertinimo tvarka (2)



Įvadas: geriausias sprendimas

Nuo senovės žmonės ieško geriausių sprendimų: kuo daugiau maisto, kuo daugiau naudos bei kuo mažiau išlaidų ar pastangų.



Matematinų metodų ir modelių tipai: 4 pagrindinės grupės

(1)

Normatyvinio ir balansinio planavimo metodai

(2)

Matematinio programavimo (optimizavimo) metodai

(3)

Tikimybių ir statistiniai metodai

(4)

Kibernetiniai metodai

(1) Normatyvinio ir balansinio planavimo metodai

Normatyvinio ir balansinio planavimo metodų pamatą sudaro normatyvų matrica. Pati normatyvų matrica ir yra matematinis modelis. Nustatomi svarbiausi ekonominiai parametrai (plėtros tempai, rentabilumo normatyvai). Šio modelio detalizacija, naudojant specialius tiesinio programavimo metodus, leidžia parinkti optimalų direktyvų įvykdymo variantą.

(2) Matematinio programavimo (optimizavimo) metodai

Matematinis programavimas yra matematinė disciplina, nagrinėjanti sąlyginių ekstremalių uždavinių sprendimo teoriją ir metodus. Analitiniai matematiniai programavimo metodai skirstomi į šakas:

tiesinį programavimą

kai apribojimai ir tikslo funkcija išreikšti tiesinėmis lygtimis ir nelygybėmis;

netiesinį programavimą

kai nors vienas apribojimas (ar tikslo funkcija) išreikštas netiesinėmis lygtimi ar nelygybe.

(2) Netiesinis programavimas

Skirstomas į iškilųjį ir neiškilųjį programavimą.

Iškilusis programavimas

tiria maksimumo radimo problemą, kai kintamųjų ryšius išreiškia funkcijos, iškilosios aukštyn (arba minimumo problemą - kai funkcija išgaubta žemyn). Iš iškilojo programavimo metodų plačiausiai yra išnagrinėti kvadratinio programavimo metodai, kur ryšius tarp kintamųjų išreiškianti iškiloji funkcija yra kvadratinė.

Neiškilojo programavimo

bendrųjų sprendimo metodų nėra sukurta; yra nemažai apytikslių metodų, naudojamų tik kai kurių uždavinių sprendimo atvejais (pvz., gradientiniai metodai, kintamos metrikos metodai).

(2) Matematinio programavimo (optimizavimo) metodai

Stochastinis programavimas

Uždavinio sąlygų apibūdinimo parametrai nėra konkrečiai apibrėžti skaičiai, o atsitiktiniai dydžiai.

Sveikaskaitis programavimas

Šio tipo uždavinių nemažai pasitaiko sprendžiant tiesnio programavimo uždavinius, kai kintamieji yra transporto priemonės, technologiniai įrenginiai, žmonės.

Dinaminis programavimas

Ieškomas optimalus sprendinys ne vienam laikotarpiui, bet keletui.

Lošimų teorija

Nagrinėja optimizavimo problemas, kai yra keli nepalyginami optimalumo kriterijai, arba sąveika kelių asmenų ar organizacijų, iš kurių kiekviena turi savo optimalumo kriterijų (konkurencija, arbitražas, kooperacija ir t.t.). Tai teorija, nagrinėjanti konfliktines situacijas.

(3) Tikimybių ir statistiniai metodai

Tikimybių ir statistiniai metodai

Tikimybių ir statistiniams metodams priklauso nemažai ekonometrinių metodų:

- regresijos metodai (tiesinė, daugialypė, logistinė);
- dispersinė analizė;
- klasterinė analizė;
- diskriminantinė analizė;
- faktorinė analizė.

Galima dar paminėti masinio aptarnavimo (laukimo arba eilių) teorijos metodus.

(4) Kibernetiniai metodai

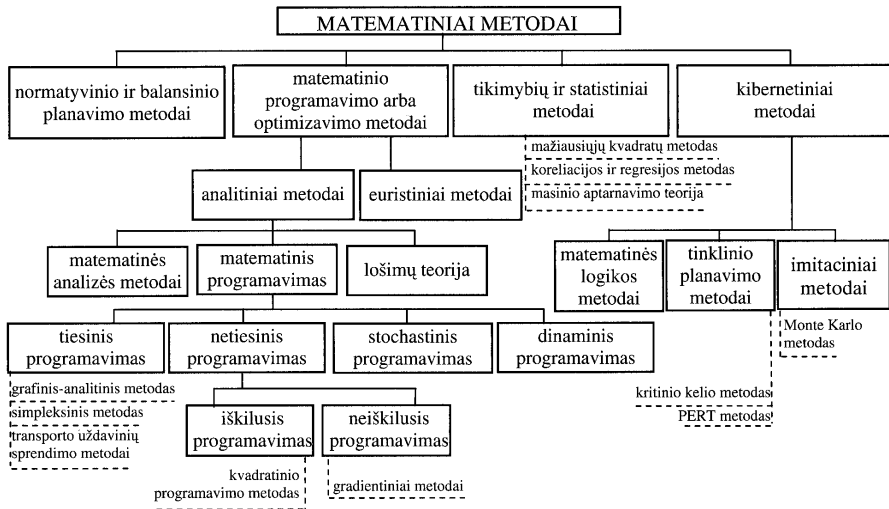
Tinklinio planavimo metodai

Tinklinio planavimo metodai naudojami didžiuliams transporto gamybos, mokslinio tyrimo, ūkinių ir finansinių operacijų kompleksams kontroliuoti ir koordinuoti.

Imitaciniai metodai

Imitaciniai metodai leidžia tirti įvairius ekonominius bei techninius reiškinius ne natūraliomis sąlygomis, bet dirbtinai sukurtose, kai imituojams nagrinėjamas reiškiny ar procesas. Modeliuojami įvairūs gamybinių sistemų variantai dar prieš juos realizuojant, ir parenkamas geriausias sprendinys. Imitacijos procesas kompiuteriu trunka keleta minučių, per kurias apskaičiuojami duomenys apie procesus, truncančius visus metus.

Matematinų metodų ir modelių schema



Praktikoje daugelio tiesinio programavimo uždavinių apribojimų matrica yra specialios struktūros. Tokiems uždaviniams spręsti sukurti paprastesni ir efektyvesni negu simplekso metodai. Pavyzdžiui, transporto uždavinys, kai ieškoma vienalyčio produkto ekonomiškiausio pervežimo plano iš siūntimo į paskirties punktus, sprendžiamas potencialų, paskyrstymo ir kt. metodais.

Transporto uždavinio formulavimas

- Tegu m siuntimo punktuose $A_1, \dots, A_m$ atitinkami yra $a_1, \dots, a_n$ vienetų kokio nors vienalyčio krovinio. Tą krovinį reikia pervežti į n paskirties punktus $B_1, \dots, B_n$, kuriems jo reikia atitinkamai $b_1, \dots, b_n$ vienetų.
- Krovinio vieneto pervežimo kaina iš siuntimo punkto A_i į paskirties punktą B_j lygi c_{ij} . Be to, krovinio pervežimo kaina tiesiai proporcinga pervežamo krovinio kiekiui.
- Reikia sudaryti tokį krovinio pervežimo planą, pagal kurį visi kroviniai iš siuntimo punktų bus išvežti, visų vartotojų poreikiai bus patenkinti, o bendroji pervežimo kaina bus mažiausia.

Transporto uždavinio lentelė (1)

Siuntimo punktai	Paskirties punktai				Ištekliai
	B_1	B_2	$\dots$	B_n	
A_1	x_{11} c_{11}	x_{12} c_{12}	$\dots$	x_{1n} c_{1n}	a_1
A_2	x_{21} c_{21}	x_{22} c_{22}	$\dots$	x_{2n} c_{2n}	a_2
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
A_m	x_{m1} c_{m1}	x_{m2} c_{m2}	$\dots$	x_{mn} c_{mn}	a_m
Poreikiai	b_1	b_2	$\dots$	b_n	

Transporto uždavinių lentelė (2)

	Swedbank	SEB	ŠB	
Antanas	5%	2%	5%	50
Leonas	25%	15%	40%	60
Petras	15%	30%	10%	10
	100	45	60	

Matematinio modelio sudarymo schema

- Įvedami kintamieji (nežinomieji)
- Užrašomi apribojimai kintamiesiems
- Užrašoma tikslo funkcija ir nurodoma *min* ar *max* jos reikšmė yra ieškoma
- Užrašomas uždavinio matematinis modelis

Transporto uždavinio matematinis modelis

$$\min \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij},$$

$$x_{11} + \dots + x_{1n} = a_1,$$

.....

$$x_{m1} + \dots + x_{mn} = a_m,$$

$$x_{11} + \dots + x_{m1} = b_1,$$

.....

$$x_{1n} + \dots + x_{mn} = b_n,$$

$$x_{ij} \geq 0, \quad i = 1, \dots, m; \quad j = 1, \dots, n.$$

Transporto uždavinio savybės (1)

1 savybė

Transporto uždavinys turi optimajųjį pervežimo planą tada ir tik tada, kai tenkinama balanso sąlyga $\sum a_i = \sum b_j$.

1 apibrėžimas

Jeigu teisinga lygybė $\sum a_i = \sum b_j$, tai transporto uždavinys vadinamas subalansuotu.

2 apibrėžimas

Jei teisinga nelygybė $\sum a_i > \sum b_j$ arba $\sum a_i < \sum b_j$, tai transporto uždavinys vadinamas nesubalansuotu.

Transporto uždavinio matricinė forma

Reikia rasti $\min \langle c, x \rangle$, kai $Ax = b$, $x \geq 0$. Čia

$$\mathbf{x} = (x_{11}, \dots, x_{1n}, \dots, x_{m1}, \dots, x_{mn})^T,$$

$$\mathbf{c} = (c_{11}, \dots, c_{1n}, \dots, c_{m1}, \dots, c_{mn})^T,$$

$$A = \begin{pmatrix} 1 & 1 & \dots & 1 & 0 & 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 1 & 1 & \dots & 1 & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & \dots & 1 & 1 & \dots & 1 \\ 1 & 0 & \dots & 0 & 1 & 0 & \dots & 0 & \dots & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & 0 & 0 & \dots & 1 & \dots & 0 & 0 & \dots & 1 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} a_1 \\ a_2 \\ \dots \\ a_m \\ b_1 \\ \dots \\ b_n \end{pmatrix}$$

$$\mathbf{a}^{ij} = (0, 0, \dots, 0, 1, 0, \dots, 0, 1, 0, \dots, 0)^T.$$

Vektorius $\mathbf{a}^{ij}$ i -oji ir m -oji koordinatės yra vienetai, o kitos - nuliai.

Transporto uždavinio savybės (2)

2 savybė

Transporto uždavinio apribojimų matricos A rangas lygus $m + n - 1$. Vadinasi, bazinių kintamųjų yra $m + n - 1$, o laisvųjų kintamųjų skaičius $mn - (m + n - 1)$.

3 savybė

Tegu ištekliai a_i ir poreikiai b_j yra sveikieji skaičiai. Tada egzistuoja bent vienas transporto uždavinio optimalusis bazinis sprendinys su sveikosiomis bazinių kintamųjų reikšmėmis.

Baziniai ir laisvieji langeliai

Transporto uždavinio lentelės langelį, kuris yra eilės A_i ir stulpelio B_j sankirtoje, žymėsime (i, j) .

3 apibrėžimas

Langelį, kuriame įrašytas bazinio sprendinio bazinis kintamasis, vadinsime baziniu, arba užpildytu langeliu.

4 apibrėžimas

Langelį, kuris atitinka bazinio sprendinio laisvąjį kintamąjį, vadinsime laisvuju, arba neužpildytu langeliu.

5 apibrėžimas

Sutvarkytas langelių rinkinys, kuriame gretimi du langeliai yra transporto uždavinio lentelės toje pačioje eilutėje arba tame pačiame stulpelyje, tačiau jokie trys langeliai nėra vienoje eilutėje arba viename stulpelyje, vadinamas grandine.

Pvz., grandine

$$(i_1, j_1), (i_1, j_2), (i_2, j_2), \dots, (i_{k-1}, j_k), (i_k, j_k)$$

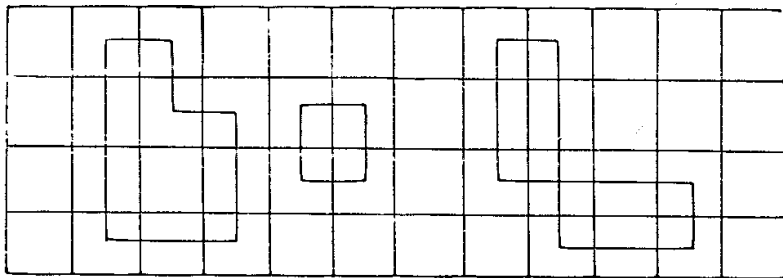
jungia siuntimo punktą A_{i_1} su paskirties punktu B_{j_k} . Vieno langelio grandinė (i_r, j_r) jungia punktą A_{i_r} su punktu B_{j_r} .

6 apibrėžimas

Grandinė, kurioje pirmasis ir paskutinis langeliai yra toje pačioje transporto uždavinio lentelės eilutėje, arba tame pačiame stulpelyje, vadinama ciklu.

Grafiškai ciklą galima vaižduoti uždara laužte, kurios viršūnės yra ciklo langeliuose. Kiekvienoje viršūnėje susiduria dvi laužtės atkarpos, kurių viena eina eilute, o kita stulpeliu.

Galimų ciklų pavyzdžiai



Transporto uždavinio savybės (3)

4 savybė

Sakykime, kad transporto uždavinio lentelėje, kurioje yra m eilučių ir n stulpelių, laisvai pažymėta $m + n$ langelių. Jei $mn \geq m + n$, tai egzistuoja ciklas, kurio viršūnės yra pažymėtuose langeliuose.

5 savybė

Sakykime, kad transporto uždavinio lentelė aprašytas pervežimo planas. Jis yra bazinis tada ir tik tada, kai iš užimtųjų langelių negalima sudaryti ciklo.

6 savybė

Sakykime, kad transporto uždavinio lentelė aprašytas pervežimo planas. Tada kiekvieną laisvąjį langelį atitinka vienas ir tik vienas ciklas, kurio pradžia yra tame langelyje, o visi kiti langeliai yra baziniai (užimtieji).

Transporto uždavinys sprendžiamas remiantis simplekso metodo idėja. Todėl turi būti patenkintos 3 salygos:

- Uždavinys duotas forma $K_{\max}$.
- Apribojimų sistemos laisvieji nariai yra neneigiami.
- Žinomas aprobojimų sistemos neneigiamas bazinis sprendinys.

Susipažinsime su dviem pradinio neneigiamo bazinio sprendinio radimo metodais:

- šiaurės vakarų kampo (įstrižainės) metodu,
- mažiausios kainos (mažiausio elemento) metodu.

Šiaurės vakarų kampo metodas

Imkime subalansuotą transporto uždavinį:

	B_1	B_2	$\dots$	B_n	a_i
A_i	c_{11}	c_{12}	$\dots$	c_{1n}	a_1
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
A_m	c_{m1}	c_{m2}	$\dots$	c_{mn}	a_m
b_j	b_1	b_2	$\dots$	b_n	

Šiaurės vakarų kampo metodas (1)

- Iš pradžių pildomas $(1, 1)$ - šiaurės vakarų kampo langelis: įrašomas $x_{11}^0 = \min \{a_1, b_1\}$.
- Jeigu $a_1 > b_1$, tai $x_{11}^0 = b_1$. Vadinasi, gavėjo B_1 poreikiai visiškai patenkinti, o siuntimo punkte A_1 liko $a_1 - b_1$ vienetų krovinio. Kadangi gavėjui B_1 iš kitų siuntimo punktų krovinio vežti nereikia, tai lentelėje išbraukiamas pirmas stulpelis.
- Jeigu $a_1 < b_1$, tai $x_{11}^0 = a_1$. Iš A_1 išvežtas visas krovinys, tačiau gavėjo B_1 poreikiai ne visiškai patenkinti. Lentelėje išbraukiama pirmoji eilutė.
- Jeigu $a_1 = b_1$, tai $x_{11}^0 = a_1 = b_1$. Iš A_1 išvežtas visas krovinys ir visiškai patenkinti gavėjo B_1 poreikiai ne visiškai patenkinti. Lentelėje išbraukiama pirmoji eilutė.

Šiaurės vakarų kampo metodas (2)

- Toliau lentelė nagrinėjama be išbrauktos eilutės arba stulpelio. Tagal tas pačias taisykles užpildomas naujas šiaurės vakarų kampas.
- Kiekvienu žingsniu užpildomas vienas langelis ir lentelėje išbraukiama viena eilutė, arba stulpelis.
- Procesas tęsiamas, kol užpildomas paskutinis (m, n) langelis. Paskutiniame žingsnyje kartu išbraukiama eilutė ir stulpelis.
- Gauname $m + n - 1$ užpildytą langelį. Neužpildytus langelius atitinkančias kintamųjų reikšmes prilyginę nuliui x_{ij}^0 , gauname pradinį pervežimo planą.

Šiaurės vakarų kampo metodu rastas pervežimo planas yra bazinis.

Rasti bazinį sprendinį šiaurės vakarų kampo metodu

	B_1	B_2	B_3	a_i
A_1	10	15	8	600
A_2	9	13	11	500
b_j	250	450	400	1100

Rasti bazinį sprendinį šiaurės vakarų kampo metodu (ats.)

	B_1	B_2	B_3	a_i
A_1	10 250	15 350	8	600
A_2	9	13 100	11 400	500
b_j	250	450	400	1100

Mažiausios kainos (mažiausio elemento) metodas

Sudarydami pradinį bazinį pervežimų planą šiaurės vakarų kampo metodu, neatsižvelgiama į krovimo pervežimo kainą. Dažnai toks planas labai nutolęs nuo optimaliojo. Todėl ieškant optimaliojo plano, reikia atlikti daug iteracijų.

Iteracijų skaičius būna mažesnis, kai pradinis planas randamas mažiausio elemento metodu. Taikant šį metodą, kiekviename žingsnyje pildomas langelis, kurį atitinka mažiausia pervežimo kaina tarp neužpildytų langelių. Po to išbraukiama eilutė arba stulpelis (pagal šiaurės vakarų kampo metodo schemą).

Rasti bazinį sprendinį mažiausio elemento metodu

	B_1	B_2	B_3	a_i
A_1	10	15	8	600
A_2	9	13	11	500
b_j	250	450	400	1100

Rasti bazinesį sprendinį mažiausio elemento metodu (ats.)

	B_1	B_2	B_3	a_i
A_1	10 200	15 400	8	600
A_2	9 250	13 250	11	500
b_j	250	450	400	1100

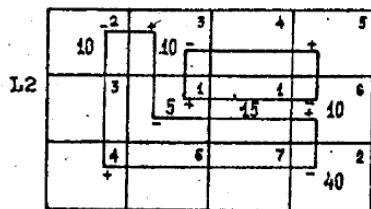
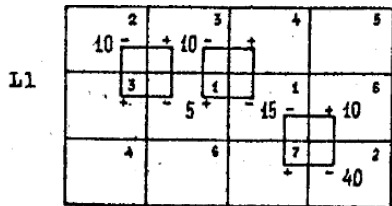
Pervežimų lentelės duoto laisvojo (neužpildyto) langelio ciklu vadinama uždara stačiakampė laužtinė linija, tenkinanti sąlygas:

- laužtės pradžia ir pabaiga yra laisvame langelyje, o kitos viršunės yra baziniuose langeliuose (nebūtinai visuose);
- kiekvienoje laužtės viršunėje sueina tik dvi grandys, kurių viena eina eilutę, o kita - stulpeliu (ciklas gali save kirsti, bet susikirtimo taškas nėra ciklo viršunė);
- ciklo viršunėms pakaitomis priskiriami $+$ ir $-$, pradedant nuo viršunės laisvame langelyje, kuriai priskiriamas $+$.

Kiekvienam laisvam langeliui egzistuoja ciklas ir tik vienas.

Pervežimų lentelėje yra $mn - (m + n - 1)$ laisvų langelių, todėl galima nubrėžti tiek ciklų.

Galimų ciklų pavyzdžiai



Lentelėje L1 pavaizduoti laisvų langelių x_{23}, x_{24}, x_{33} ciklai, o L2 pavaizduoti x_{24}, x_{34} ciklai.

7 apibrėžimas

Poslinkis ciklu per skaičiu x - tai veiksmas, kada prie skaičių, esančių teigiamose ciklo viršūnėse, pridedamas skaičius x , iš skaičių, esančių neigiamose viršūnėse, atimamas x , o skaičiai nesantys ciklo viršūnėse nekeičiami.

Lentelėje L3 pavaizduotas langelio x_{32} ciklas, o L4 - poslinkio x_{32} ciklu per 5 rezultatas

L3

	2	3	4	5
10	10			
	3	1	1	6
		5	15	10
		6	7	
	4			2
				40

L4

	2	3	4	5
10	10			
	3	1	1	6
			15	15
	4	6	7	2
		5		35

Transporto uždavinio sprendimas paskirstymo metodu (1)

Tarkime, kad žinomas subalansuoto transporto uždavinio bazinis planas x^0 . Tikslų funkcija taške x^0 įgyja reikšmę

$$f(x^0) = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij}^0.$$

Imkime bet kurį laisvąjį langelį (i_0, j_0) ir, priskyre jam $+$, sudarykime ženklintą ciklą. Apskaičiuoju

$$\Theta = \min_{(i,j)^-} x_{ij}^0$$

ir atlikę poslinkį cikle skaičiumi Θ , gauname kitą atraminį planą x^1 . Tikslų funkcija taške x^1 įgyja reikšmę

$$f(x^1) = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij}^1.$$

Transporto uždavinio sprendimas paskirstymo metodu (2)

Pokytį galima išreikšti

$$\Delta f = f(x^1) - f(x^0) = \gamma_{i_0 j_0} \Theta,$$

kur

$$\gamma_{i_0 j_0} = \sum_{(i,j)^+} c_{ij} - \sum_{(i,j)^-} c_{ij}$$

yra laisvojo langelio (i_0, j_0) skaitinė charakteristika. Čia

$$\sum_{(i,j)^+} c_{ij}, \sum_{(i,j)^-} c_{ij}$$

sumos pervežimo kainų, esančių atitinkamai pluso ir minuso ženklais pažymėtuose ciklo langeliuose.

Paskirstymo metodo realizavimo schema

- 1 Šiaurės vakarų kampo ar mažiausio elemento metodu randamas pradinis bazinis planas x^0 .
- 2 Kiekvienam laisvajam langeliui (i, j) sudaromas ženklintas ciklas ir apskaičiuojama skaitmeninė charakteristika γ_{ij} . Jeigu visos $\gamma_{ij} \geq 0$, tai bazinis planas x^0 yra optimalus.
- 3 Randama mažiausia neigiama charakteristika

$$\gamma_{i_0, j_0} = \min \gamma_{ij} < 0.$$

Langelio (i_0, j_0) ženklinto ciklo pagalba randamas

$$\Theta = \min_{(i, j) \in \text{ciklas}} x_{ij}^0.$$

Atlikę poslinkį šiame cikle skaičiumi Θ , gauname kitą lentelę ir ją atitinkantį atraminį planą x^1 , it t.t.

Įsprendžiant paskirstymo metodą

Sandėliai	B_1	B_2	B_3	B_4	B_5	Atsargos
A_1	10	6	7	8	5	45
A_2	8	7	9	4	4	20
A_3	10	9	7	6	5	55
A_4	4	5	6	10	8	50
Paraiškos	20	20	30	40	60	170

lšspręsti paskirstymo metodu (ats.)

Optimalus sprendinys: 880 eu

Sandėliai	B_1	B_2	B_3	B_4	B_5	Atsargos
A_1	10 20	6	7	8	5 25	45
A_2	8	7	9	4 20	4	20
A_3	10	9	7	6 20	5 35	55
A_4	4 20	5	6 30	10	8	50
Paraiškos	20	20	30	40	60	170

Potencialų metodas (1)

Tiesioginis transporto uždavinys:

$$\min \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij}$$

$$\left\{ \begin{array}{lll} x_{11} + x_{12} + \dots + x_{1n} & = a_1 & u_1 \\ x_{21} + x_{22} + \dots + x_{2n} & = a_2 & u_2 \\ \dots & \dots & \dots \\ x_{m1} + x_{m2} + \dots + x_{mn} & = a_m & u_m \\ x_{11} + x_{21} + \dots + x_{m1} & = b_1 & v_1 \\ x_{12} + x_{22} + \dots + x_{m2} & = b_2 & v_2 \\ \dots & \dots & \dots \\ x_{1n} + x_{2n} + \dots + x_{mn} & = b_n & v_n \end{array} \right.$$

$$\forall x_{ij} \geq 0.$$

Potencialų metodas (2)

Dualusis transporto uždavinys:

$$\max \left(\sum_{i=1}^m a_i u_i + \sum_{j=1}^n b_j v_j \right)$$

$$\left\{ \begin{array}{ll} u_1 + v_1 & \leq c_{11} \\ u_1 + v_2 & \leq c_{12} \\ \dots & \dots \\ u_1 + v_n & \leq c_{1n} \\ u_2 + v_1 & \leq c_{21} \\ \dots & \dots \\ u_m + v_n & \leq c_{mn} \end{array} \right.$$

$$\forall u_i \in \mathbb{R}, \quad \forall v_j \in \mathbb{R}.$$

Potencialai (1)

Tarkime, kad $\bar{x}_{kl} > 0$. Tada gauname $m + n - 1$ lygčių sistemą su $m + n$ nežinomaisiais:

$$\left\{ \begin{array}{ll} \bar{u}_1 + \bar{v}_1 & = c_{11} \\ \bar{u}_1 + \bar{v}_2 & = c_{12} \\ \dots\dots\dots & \dots \\ \bar{u}_m + \bar{v}_n & = c_{mn} \end{array} \right.$$

Ši sistema turi be galo daug sprendinių. Norėdami gauti konkretų sprendinį, pasirenkame laisvai kurio nors kintamojo reikšmę, pvz., $\bar{u}_1 = 0$. Tuomet kitų nežinomųjų reikšmės surasime vienareikšmiškai. Rastos reikšmės vadinamos punktų *potencialais*: $\bar{u}_k$ - siuntimo punktų potencialai, $\bar{v}_l$ - gavimo punktų potencialai.

Langelio i, j charakteristika (įvertinimas) apskaičiuojamas pagal formulę

$$\gamma_{i,j} = \bar{u}_i + \bar{v}_j - c_{ij}.$$

Jeigu visi įvertinimai yra neteigiami, tai bazinis planas yra optimalus. Priešingu atveju parenkame didžiausią teigiamą įvertinimą:

$$\gamma_{i_0,j_0} = \max_{\gamma_{i,j} > 0} \gamma_{i,j} = \max \{ \bar{u}_i + \bar{v}_j - c_{ij} \}.$$

Potencialų metodo realizavimo schema

- 1 Šiaurės vakarų kampo ar mažiausio elemento metodu randamas pradinis bazinis planas x^0 .
- 2 Išsprendę sistemą, randame potencialų reikšmes. Kiekvienam laisvajam langeliui (i, j) apskaičiuojamas įvertinimas:

$$\gamma_{i,j} = \bar{u}_i + \bar{v}_j - c_{ij}.$$

Jeigu $\forall \gamma_{i,j} \leq 0$, tai bazinis planas x^0 yra optimalus.

- 3 Pagal didžiausią teigiamą įvertinimą

$$\gamma_{i_0, j_0} = \max \gamma_{ij} > 0$$

parenkame laisvąjį langelį. Priskyre langeliui (i_0, j_0) pliuso ženklą, randame

$$\Theta = \min_{(i,j)^-} x_{ij}^0.$$

Atlikę poslinkį šiame cikle skaičiumi Θ , gauname kitą lentelę ir ją atitinkantį atraminį planą x^1 , it t.t.

Išspręsti potencialų metodą

	B_1	B_2	B_3	a_i
A_1	10	15	8	600
A_2	9	13	11	500
b_j	250	450	400	1100

Išspręsti potencialų metodą (ats.)

	B_1	B_2	B_3	a_i
A_1	10 250	15 350	8	600
A_2	9 100	13 400	11	500
b_j	250	450	400	1100

Transporto uždavinys yra išsigimęs, jeigu bent vieno jo bazinio plano bazinio kintamojo reikšmė lygi nuliui. Sprendžiant išsigimusį uždavinį, gali būti pradinis bazinis planas jau išsigimęs arba neišsigimęs, tačiau kitoje iteracijoje gaunamas išsigimęs bazinis planas.

Kaip kiekvienam išsigimusiam baziniam planui parinkti papildomus užimtuosius langelius, kad jų skaičius būtų lygus $m + n - 1$?

(1) Pradinis bazinis planas yra išsigimęs

Išsigimęs bazinis pradinis planas gaunamas tada, kai užpildant eilinį langelį, jį atitinkančio siuntimo punkto krovinio ištekliai ir paskirties punkto poreikiai lygūs nuliui. Tada į tą langelį įrašome nulį, t.y. laukome jį užimtu, ir išbraukiame eilutę, kurioje jis yra. Taip parinkdami užimtuosius langelius, gausime pradinį bazinį planą, kurį atitinkančioje lentelėje yra $m + n - 1$ užimtas langelis.

(2) Pradinis bazinis planas nėra išsigimęs

Antrąjį atvejį gauname, kai ženklinto ciklo minuso ženklu pažymėtuose langeliuose yra dvi arba daugiau mažiausios (vienodos) bazinių kintamųjų reikšmės. Atlikę poslinkį tokiu ciklu, gauname du arba daugiau langelių, kuriuose atsiranda nuliui, t.y. langeliai tampa laisvi. Vieną iš tų langelių paliekame laisvą, o į kitus įrašome nulius ir juos laikome užimtaisiais.

Išspręsti potencialų metodu ($z_{min} = 1900$)

Siuntimo punktai	Paskirties punktai				Ištekliai
	B_1	B_2	B_3	B_4	
A_1	3	6	5	1	100
A_2	1	4	3	2	300
A_3	3	3	1	2	600
Poreikiai	300	400	100	200	1000

Nesubalansuotas transporto uždavinys (1)

Apibrėžimas

Jeigu transporto uždavinio bendrieji ištekliai $\sum a_i$ viršija bendruosius poreikius $\sum b_j$, arba atvirkščiai, tai transporto uždavinys vadinamas nesubalansuotu.

Nesubalansuotas transporto uždavinys (2)

Jeigu $\sum a_i > \sum b_j$ ir nesvarbu, iš kurių siuntimo punktų nebus išvežtas krovinys, tai papildę transporto uždavinį fiktyviu gavėjų B_{n+1} , kurio poreikiai yra

$$b_{n+1} = \sum_{i=1}^m a_i - \sum_{j=1}^n b_j$$

vienetu krovinio, ir laikydami pervežimo kainą iš bet kurio siuntimo punkto į fiktyvų paskirties punktą B_{n+1} lygia nuliui, gauname subalansuotą transporto uždavinį.

Nesubalansuotas transporto uždavinys (3)

Jeigu $\sum a_i < \sum b_j$ ir nesvarbu, kurio paskirties punkto poreikiai nebus visiškai patenkinti, tai papildę transporto uždavinį fiktyviu siuntėju A_{m+1} , kurio ištekliai yra

$$a_{m+1} = \sum_{j=1}^n b_j - \sum_{i=1}^m a_i$$

vienetu krovinio, ir laikydami pervežimo kainą iš fiktyvaus siuntėjo į bet kurį paskirties punktą lygia nuliui, gauname subalansuotą transporto uždavinį.

Nesubalansuotas transporto uždavinys (4)

Nesubalansuotų ir juos atitinkančių subalansuotų uždavinių tikslo funkcijos yra vienodos.

Pradinį bazinį planą sudarant mažiausio elemento metodu, iš pradžių reikia pildyti langelius su teigiamomis kainomis. Paskiausiai pildomi langeliai, esantys papildomoje eilutėje arba stulpelyje. Tada subalansuotą uždavinį sprendžiant potencialų metodu, reikia atlikti mažiau iteracijų.

Nesubalansuotas transporto uždavinys (5)

Radę subalansuoto uždavinio optimalųjį pervežimo planą, galime užrašyti atitinkamo nesubalansuoto uždavinio optimalųjį planą.

Krovinio kiekis, kurį pagal subalansuoto uždavinio optimalųjį planą reikia nuvežti iš fiktyvaus punkto A_{m+1} į paskirties punktus B_j , reiškia, kiek vienetų krovinio nebus atvežta šiam gavėjui.

Krovinio kiekis, kurį pagal optimalųjį pervežimo planą reikia nuvežti iš siuntimo punkto A_i į fiktyvų paskirties punktą B_{n+1} , reiškia, kiek vienetų krovinio nebus išvežta iš siuntimo punkto A_i .

Išspręsti potencialų metodu ($z_{min} = 193$)

Siuntimo punktai	Paskirties punktai				Ištekliai
	B_1	B_2	B_3	B_4	
A_1	5	3	6	4	26
A_2	4	1	2	1	36
A_3	3	2	4	1	36
Poreikiai	35	37	30	27	$129 > 98$

Tinkliniu planavimu vadinamas susijusių darbų (operacijų) komplekso realizavimo tyrimo metodas. Taikant šį metodą, darbų kompleksas vaizduojamas schema, vadinama tinkliniu grafiku.

Remiantis tuo grafiku, sudaroma darbų atlikimo eilė, apskaičiuojamas projekto realizavimo laikas, sprendžiami su projekto įgyvendinimu susiję uždaviniai.

Grafai ir tinklai

Grafą galima apibrėžti kaip aibių porą (S, U) , kur S - viršūnių aibė, U - briaunų aibė. Toliau naginėsime tik orientuotus grafus.

Kelias (grandinė)

Grafo briaunų seka $\{(P_{i_1}, P_{i_2}), (P_{i_2}, P_{i_3}), \dots, (P_{i_k}, P_{i_{k+1}})\}$, kurioje kiekvienos paskesnės briaunos pradžia sutampa su ankstesnės pabaiga, vadinama keliu, jungiančiu viršūnę P_{i_1} su viršūne $P_{i_{k+1}}$.

Ciklas

Kelias, kuris prasideda ir baigiasi toje pačioje viršūnėje, vadinamas ciklu. Vienos briaunos ciklas vadinamas kilpa.

Tinklas

Grafas, kurio elementams (dažniausiai briaunoms) priskiriami skaičiai, vadinamas tinklu.

Sudetingame darbų komplekse daug darbų, susijusių technologiniais ryšiais. Viena darbą negalima pradėti, nebaigus kitų. Sudarant tokio darbų komplekso realizavimo projektą, patogų tą darbų kompleksą vaizduoti grafu, kuris vadinamas tinkliniu grafiku.

Norint sudaryti tinklinį grafiką, reikia žinoti visus darbus, jų atlikimo eilę ir kiekvieno darbo trukmę. Pradinė informacija gali būti pateikta įvairiai.

$$1.1 = 1.2$$

1.1 lentelė

Darbas	Darbai, baigti prieš šį darbą	Darbo trukmė
D_1	—	4
D_2	—	3
D_3	—	2
D_4	D_1, D_2	4
D_5	D_2, D_3	7
D_6	D_2	5
D_7	D_4, D_6	7
D_8	D_4, D_6	9
D_9	D_5	5
D_{10}	D_8	4

1.2 lentelė

Darbas	Darbai, atliekami po šio darbo	Darbo trukmė
D_1	D_4	4
D_2	D_4, D_5, D_6	3
D_3	D_5	2
D_4	D_7, D_8	4
D_5	D_9	7
D_6	D_7, D_8	5
D_7	—	7
D_8	D_{10}	9
D_9	—	5
D_{10}	—	4

Tinkliniame grafike kiekvienas darbas vaizduojamas kryptine atkarpa (briauna) (P_i, P_j) . Atkarpos pradžia reiškia darbo pradžią, o pabaiga – darbo pabaigą. Grafo viršūnės P_i vadinamos įvykiais. Kiekvienas įvykis nusako vieno darbo pradžią, kitų pabaigą. Jeigu P_i ir P_j yra to paties darbo pradžios ir pabaigos įvykiai, tai sakoma, kad juos jungia darbas (P_i, P_j) . Laikysime, kad visi tinkliniame grafike vaizduojami darbai yra elementarūs, t. y. darbai, kuriuos galima pradėti tik tada, kai visiškai užbaigti ankstesni darbai.

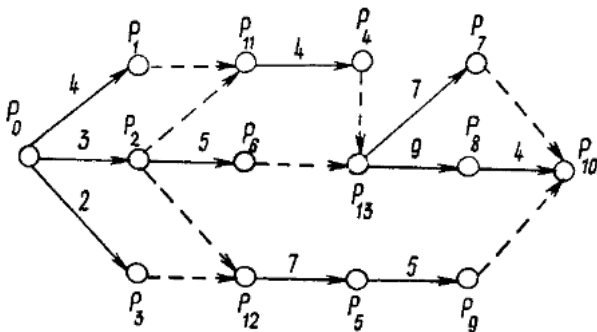
Tinklinio grafiko sudarymo schema (1)

Tinklinį grafiką galime sudaryti pagal šitokią schemą:

1. Kiekvienam darbui D_j priskiriame pabaigos įvykį P_j ;
2. Jeigu prieš kurį nors darbą nėra darbų, kurie turi būti užbaigti, tai jo pradžios įvykį žymėsime P_0 ir vadinsime projekto pradžia;
3. Jeigu prieš darbą D_j reikia atlikti tik vieną darbą D_i , tai darbo D_j pradžios įvykiu laikysime darbo D_i pabaigos įvykį;
4. Jeigu prieš darbą D_j reikia atlikti darbus $D_{i_1}, D_{i_2}, \dots, D_{i_k}$, kai $k > 1$, tai tinklinį grafiką papildome nauju įvykiu P_{j1} , kurį laikysime darbo D_j pradžia. Darbų $D_{i_1}, D_{i_2}, \dots, D_{i_k}$ pabaigos įvykius su įvykiu P_{j1} sujungiame fiktyviais nulinės trukmės darbais. Fiktyvūs darbai grafiškai vaizduojami punktyrinėmis linijomis

5. Darbo, po kurio nėra kitų darbų, pabaigos įvykį vadinsime projekto pabaiga. Jeigu tinkliniame grafike yra kelios projekto pabaigos, tai jas sujungus fiktyviais darbais, liks viena pabaiga

Tinklinio grafiko sudarymo schema (2)



Sudaryti 1.1 Int. darbų komplekso tinklinį grafiką (1)

Sudarant tinklinį grafiką, patogiu naudotis lentele, gauta papildžius 1.1 lentelę darbų pradžios bei pabaigos įvykiais (1.3 lentelė). Naudojantis šia lentele, lengva išvardinti visas tinklinio grafiko briaunas – realius ir fiktyvius darbus – bei viršūnes – darbų pradžios ir pabaigos įvykius. Realųjų darbų pradžios yra trečioje grafoje, o pabaigos – antroje grafoje.

Sudaryti 1.1 Int. darbų komplekso tinklinį grafiką (2)

1.3 lentelė

Darbas	Pabaigos įvykis	Pradžios įvykis	Darbai, baigti prieš šį darbą	Darbo trukmė
D_1	P_1	P_0	—	4
D_2	P_2	P_0	—	3
D_3	P_3	P_0	—	2
D_4	P_4	P_{11}	D_1, D_2	4
D_5	P_5	P_{12}	D_2, D_3	7
D_6	P_6	P_2	D_2	5
D_7	P_7	P_{13}	D_4, D_6	7
D_8	P_8	P_{13}	D_4, D_6	9
D_9	P_9	P_5	D_5	5
D_{10}	P_{10}	P_8	D_8	4

Sudaryti 1.1 Int. darbų komplekso tinklinį grafiką (3)

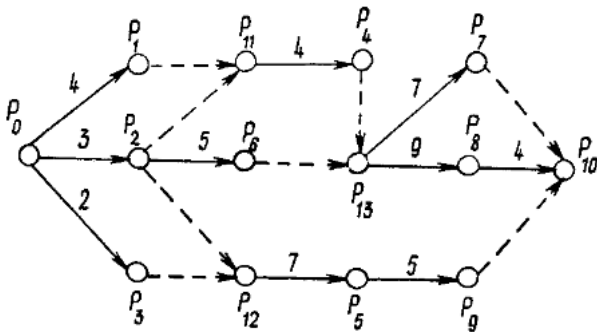
Trečiosios grafos įvykiai, kurių nėra antroje grafoje (pvz. P_{12}), yra fiktyvių darbų pabaigos įvykiai. Tokie fiktyvūs darbai prasideda pabaigos įvykiais darbų, surašytų tos pačios eilutės dešinėje pusėje. 1.4 lentelėje surašyti visi realūs ir fiktyvūs darbai, jų trukmės bei pradžios ir pabaigos įvykiai. Taigi 1.4 lentelėje aprašytas 1.1 lentelės darbų komplekso tinklinis grafikas.

1.1 Int. darbų komplekso tinklinis grafikas

1.4 lentelė

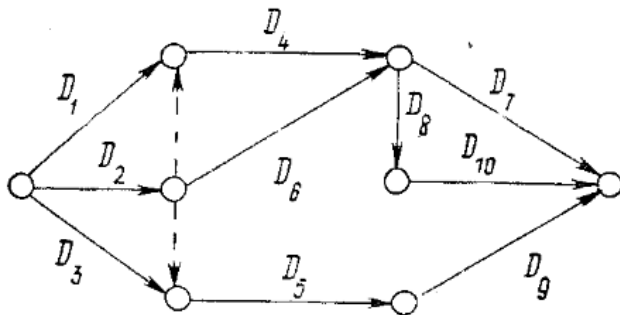
Darbas		Darbo trukmė	Darbas		Darbo trukmė
Pradžios įvykis	Pabaigos įvykis		Pradžios įvykis	Pabaigos įvykis	
P_0	P_1	4	P_8	P_{10}	4
P_0	P_2	3	P_1	P_{11}	0
P_0	P_3	2	P_2	P_{11}	0
P_{11}	P_4	4	P_2	P_{12}	0
P_{12}	P_5	7	P_3	P_{12}	0
P_2	P_6	5	P_4	P_{13}	0
P_{13}	P_7	7	P_6	P_{13}	0
P_{13}	P_8	9	P_7	P_{10}	0
P_5	P_9	5	P_9	P_{10}	0

1.1 Int. darbų komplekso tinklinis grafikas: daug fiktyvių darbų



1.1 Int. darbų komplekso tinklinis grafikas (alternatyva)

Fiktyvių darbų galėtų būti mažiau sudarant tinklinį grafiką kitu būdu:



1. Jeigu įvykį P_i su įvykiu P_j jungia daugiau kaip vienas darbas, tai tinklinis grafikas papildomas naujais įvykiais, kurie su P_j sujungiami fiktyviais darbais
2. Jeigu įvykiu P_i baigiasi ir prasideda daugiau kaip vienas darbas ir ne visiems prasidedantiems įvykiu P_i darbams būtina užbaigti visus darbus, kurie baigiasi tuo įvykiu, tai tinklinis grafikas papildomas naujais įvykiais ir fiktyviais darbais
3. Jeigu yra kelios projekto pradžios arba pabaigos, tai jos sujungiamos fiktyviais darbais

Toliau nagrinėsime tinklinių grafikų pavyzdžius, aprašytus lentele arba pavaizduotus paveiksle. Patogu nagrinėti, kai tinkliniame grafike kiekvieno darbo (P_i, P_j) pradžios įvykio numeris i mažesnis už pabaigos įvykio numerį j . Jeigu tinkliniame grafike yra darbų (P_i, P_j) , kai $i > j$, tai prieš pradėdami jį nagrinėti iš naujo sunumeruojame įvykius.

Numeravimo schema (1)

1. Pradžios įvykiui priskiriame numerį 0;
2. Tinkliniame grafike išbraukiame visas, prasidedančias įvykiu P_0 , briaunas. Jas išbraukus, atsiranda įvykių, kuriais nesibaigia jokie darbai. Tokius įvykius vadinsime *1-ojo rango įvykiais*. Pažymėsime, kad maksimalus įeinančių į kelią, jungiantį pradžios įvykį P_0 su bet kuriuo 1-ojo rango įvykiu, briaunų skaičius lygus 1;

Numeravimo schema (2)

3. Išbraukiame visas, išeinančias iš 1-ojo rango įvykių, briaunas. Įvykius, kuriais nesibaigia jokie darbai, vadinsime *2-ojo rango įvykiais*. Kelyje, jungiančiame pradžios įvykį P_0 su bet kuriuo 2-ojo rango įvykiu, maksimalus briaunų skaičius lygus 2;

4. Braukydami briaunas toliau, vieninteliu būdu visiems įvykiams priskiriame rangus. Sakome, kad tinklinį grafiką *išsluoksniuojame*. Pažymėsimė, kad maksimalus briaunų skaičius kelyje, jungiančiame pradžios įvykį P_0 su bet kuriuo r -ojo rango įvykiu, lygus r ;

5. Įvykius numeruojame šitaip: 1-ojo rango įvykiams skiriame numerius $1, 2, \dots, k_1$; čia k_1 — pirmojo rango įvykių skaičius; 2-ojo rango įvykiams priskiriame $k_1 + 1, k_1 + 2, \dots, k_1 + k_2$ ir t. t.

Taip sunumeravus tinklinio grafiko įvykius, bet kurios briaunos (darbo) (P_i, P_j) pradžios įvykis P_i bus mažesnio rango negu įvykis P_j . Vadinas, ir jo numeris i bus mažesnis negu j .

Fordo algoritmas (1)

Kai tinklinis grafikas didelis, grafiškai jį vaizduoti nepatogu. Todėl išsluoksniuojant naudojamas algoritmas. Pateiksime vieną iš labiausiai paplitusių algoritmų – vadinamąjį Fordo algoritmą, kuris taikomas ne tik tinkliniam grafikui išsluoksniuoti, bet ir jam nagrinėti.

Sakysime, kad tinklinis grafikas, kuriame yra $n+1$ įvykis $P_j, j=0, 1, \dots, n$, aprašytas lentelė. Taikydami Fordo algoritmą, laikomės šių taisyklių:

1. Kiekvienam įvykiui P_j priskiriame skaičių $\lambda_j^{(0)}=0$. Imame $k=1$ ir taikome 2 taisyklę;

2. Iš lentelės, kuria aprašytas tinklinis grafikas, bet kuriuo būdu (pavyzdžiui, pagal eilę) imame įvykius P_j ir darbus (P_i, P_j) . Įvykiui P_0 priskiriame skaičių $\lambda_0^{(k)}=0$, kitiems įvykiams – skaičius

$$\lambda_j^{(k)} = \max_i \{ \lambda_i^{(l)} + 1 \}, \quad j=1, \dots, n;$$

čia $\lambda_i^{(l)}$ – paskutinis įvykiui P_i priskirtas skaičius, t. y. $\lambda_i^{(l)} = \lambda_i^{(k-1)}$ arba $\lambda_i^{(l)} = \lambda_i^{(k)}$;

3. Lyginame įvykiui P_j priskirtus skaičius $\lambda_j^{(k-1)}$ ir $\lambda_j^{(k)}$. Jeigu

$$\lambda_j^{(k)} = \lambda_j^{(k-1)}, \quad j=0, 1, \dots, n,$$

tai taikome 4 taisyklę. Jeigu bent vienam įvykiui P_j priskirti skaičiai $\lambda_j^{(k-1)}$ ir $\lambda_j^{(k)}$ skiriasi, tai k keičiame $k+1$ ir taikome 2 taisyklę;

4. Įvykiams P_j , $j=1, \dots, n$, priskiriame rangus, kurie lygūs

$$\lambda_j^* = \lambda_j^{(k)} = \lambda_j^{(k-1)}, \quad j=1, \dots, n;$$

5. Numeruojame įvykius. Įvykio P_0 numerio nekeičiame; 1-ojo rango įvykiams priskiriame numerius $1, 2, \dots, k_1$; 2-ojo rango įvykiams – numerius $k_1+1, k_1+2, \dots, k_1+k_2$ ir t. t. Čia k_r – r -ojo rango įvykių skaičius.

Fordo algoritmo pritaikymo 1.4 Int. rezultatas

1.5 lentelė

P_j	$\lambda_j^{(0)}$	$\lambda_j^{(1)}$	$\lambda_j^{(2)}$	$\lambda_j^{(3)}$	$\lambda_j^{(4)}, \lambda_j^*$	Naujas įvykio Nr.
P_0	0	0	0	0	0	0
P_1	0	1	1	1	1	1
P_2	0	1	1	1	1	2
P_3	0	1	1	1	1	3
P_4	0	1	3	3	3	7
P_5	0	1	3	3	3	8
P_6	0	2	2	2	2	4
P_7	0	1	4	5	5	11
P_8	0	1	4	5	5	12
P_9	0	2	4	4	4	9
P_{10}	0	3	5	6	6	13
P_{11}	0	2	2	2	2	5
P_{12}	0	2	2	2	2	6
P_{13}	0	3	4	4	4	10

1. Sudarykite darbų kompleksų tinklinius grafikus
2. Nauduodamiesi Fordo algoritmu, išsluoksniuokite sudarytus tinklinius grafikus.

Darbas	Darbai, baigti prieš šį darbą	Darbo trukmė
D_1	—	3
D_2	D_1	4
D_3	D_1	5
D_4	—	6
D_5	—	5
D_6	D_3, D_4	7
D_7	D_3, D_4	5
D_8	D_2, D_6	3
D_9	D_5	8

Darbas	Darbai, baigti prieš šį darbą	Darbo trukmė
D_1	—	4
D_2	—	5
D_3	D_2	3
D_4	D_1, D_3	7
D_5	—	9
D_6	D_5	2
D_7	D_2	5
D_8	D_2	3
D_9	D_6, D_7	4

Norint apskaičiuoti projekto realizavimo laiką arba išspręsti kitus su projektu susijusius uždavinius, reikia žinoti tinklinio grafiko parametrus.

1. Kritinio kelio ir kritinio laiko sąvokos. Nagrinėsime tinklinį grafiką, kurio briaunos (P_i, P_j) vaizduoja darbus, o viršūnės $P_j, j=0, 1, \dots, n$, – darbų pradžios ar pabaigos įvykius. Kiekvienai briaunai (P_i, P_j) priskirtas skaičius t_{ij} – vaizduojamojo darbo trukmė. Skaičių t_{ij} vadinsime *briaunos (P_i, P_j) ilgiu*. Kelio $\{(P_{i_1}, P_{i_2}), (P_{i_2}, P_{i_3}), \dots, (P_{i_{k-1}}, P_{i_k}), (P_{i_k}, P_{i_{k+1}})\}$ ilgiu vadinsime į jį įeinančių briaunų ilgių sumą

$$t_{i_1 i_2} + t_{i_2 i_3} + \dots + t_{i_k i_{k+1}}.$$

Kritinis kelias ir kritinis laikas (1)

Norėdami apskaičiuoti viso projekto realizavimo trukmę, turime nustatyti kiekvieno įvykio anksčiausiąjį laiką. Laikysime, kad projekto pradžia (įvykio P_0 anksčiausias laikas) $T_0=0$. Tada anksčiausias įvykio P_j laikas, kurį žymėsime T_j^0 , lygus ilgiausio kelio iš P_0 į P_j ilgiui.

1 apibrėžimas. *Kritiniu laiku T_n vadinamas projekto pabaigos įvykio P_n anksčiausias laikas T_n^0 .*

Iš šio apibrėžimo išplaukia, kad viso projekto realizavimo trukmė lygi kritiniam laikui T_n .

2 apibrėžimas. *Bet kuris kelias iš P_0 į P_n , kurio ilgis lygus T_n , vadinamas kritiniu.*

Kritinis kelias ir kritinis laikas (2)

Tinkliniame grafike gali būti keletas kritinių kelių. Darbai, esantys kritiniame kelyje, vadinami kritiniais, o kiti — nekritiniais. Jeigu, didindami resursų kiekį (išskirdami daugiau darbo jėgos, įrengimų ar žaliavų), kritinius darbus atliksime greičiau, tai projekto realizavimo trukmė (kritinis laikas) sumažės. Tačiau, jeigu užtesime kuriuos nors kritinius darbus daugiau negu numatyta projekte, tai viso projekto realizavimo laikas pailgės. Todėl ilgiausias kelias ir pavadintas kritiniu. Sutrumpinus nekritinių darbų atlikimo trukmę, kritinis laikas nesumažės. Taupant resursus, galima užtęsti arba vėliau pradėti nekritinius darbus. Tačiau, kad viso projekto realizavimo trukmė nepadidėtų, nekritinių darbų atlikimo laiką galima prailginti tiek, kad tie darbai taptų kritiniais.

Taigi kritinis laikas gali nepakisti, nors kai kurie P_j įvyks vėliau už T_j^0 . Vėliausiąjį įvykio P_j laiką, kuris nepadidina kritinio laiko, žymėsime T_j^1 . Nesunku suvokti, kad T_j^1 lygus kritinio laiko T_n ir ilgiausio kelio iš P_j į P_n ilgio skirtumui.

Algoritmas parametrams T_j^0 apskaičiuoti

1. Paeiliui imame įvykius P_j ir darbus (P_i, P_j) . Įvykiui P_0 priskiriame skaičių $\lambda_0=0$, kitiems įvykiams P_j , $j=1, \dots, n$, – skaičius

$$\lambda_j = \max_i \{ \lambda_i + t_{ij} \}, \quad j=1, \dots, n;$$

2. Prie skaičiaus λ_j rašome įvykį (įvykius), su kurio skaičiumi λ_i teisinga (2.1) lygybė;

3. Skaičius λ_j lygus ilgiausio kelio iš P_0 į P_j ilgiui. Taigi $T_j^0 = \lambda_j$, $j=0, 1, \dots, n$.

Kritinis laikas $T_n = \lambda_n$.

Remiantis pagal 2-ąją taisyklę surašytais įvykiais, nesunku rasti visus kritinius kelius.

Algoritmas parametrams T_j^1 apskaičiuoti

1. Atvirkščia tvarka imame įvykius $\bar{P}_i$ ir darbus (P_i, P_j) . Įvykiui P_n priskiriame skaičių $\mu_n=0$, kitiems įvykiams P_i – skaičius

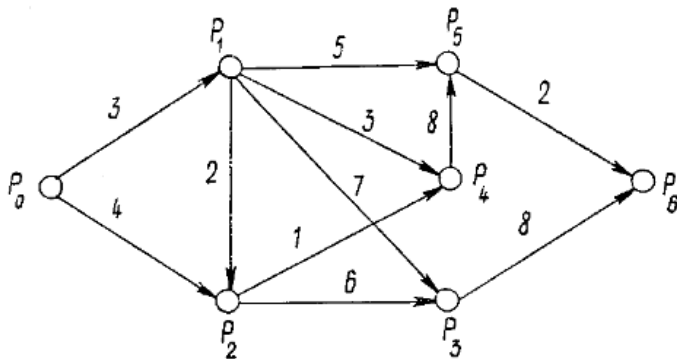
$$\mu_i = \max_j \{ \mu_j + t_{ij} \}, \quad i = n-1, \dots, 0;$$

2. Prie skaičiaus μ_i rašome įvykį (įvykius), su kurio skaičiumi μ_j teisinga (2.2) lygybė;

3. Skaičius μ_i lygus ilgiausio kelio iš P_i į P_n ilgiui. Taigi $T_n = \mu_0$, $T_i^1 = T_n - \mu_i$, $i=0, 1, \dots, n$.

Visus kritinius kelius galima rasti ir remiantis prie μ_i surašytais įvykiais.

Užd. Tinkliniu grafiku rasti parametrus T_j^0 , T_j^1 ir kritinį kelią



Naudodamiesi pirmuoju algoritmu, apskaičiuojame

$$\lambda_0 = 0;$$

$$\lambda_1 = \max \{ \lambda_0 + t_{01} \} = 3, P_0;$$

$$\lambda_2 = \max \{ \lambda_0 + t_{02}, \lambda_1 + t_{12} \} = 5, P_1;$$

$$\lambda_3 = \max \{ \lambda_1 + t_{13}, \lambda_2 + t_{23} \} = 11, P_2;$$

$$\lambda_4 = \max \{ \lambda_1 + t_{14}, \lambda_2 + t_{24} \} = 6, P_1, P_2;$$

$$\lambda_5 = \max \{ \lambda_1 + t_{15}, \lambda_4 + t_{45} \} = 14, P_4;$$

$$\lambda_6 = \max \{ \lambda_3 + t_{36}, \lambda_5 + t_{56} \} = 19, P_3.$$

Pagal 3-ąją taisyklę $T_0^0 = 0, T_1^0 = 3, T_2^0 = 5, T_3^0 = 11, T_4^0 = 6, T_5^0 = 14, T_6^0 = 19.$

Naudodamiesi antruoju algoritmu, apskaičiuojame:

$$\mu_6 = 0;$$

$$\mu_5 = \max \{ \mu_6 + t_{56} \} = 2, P_6;$$

$$\mu_4 = \max \{ \mu_5 + t_{45} \} = 10, P_5;$$

$$\mu_3 = \max \{ \mu_6 + t_{36} \} = 8, P_6;$$

$$\mu_2 = \max \{ \mu_4 + t_{24}, \mu_3 + t_{23} \} = 14, P_3;$$

$$\mu_1 = \max \{ \mu_5 + t_{15}, \mu_4 + t_{14}, \mu_3 + t_{13}, \mu_2 + t_{12} \} = 16, P_2;$$

$$\mu_0 = \max \{ \mu_2 + t_{02}, \mu_1 + t_{01} \} = 19, P_1.$$

Pagal 3-ąją taisyklę $T_6 = 19, T_0^1 = 0, T_1^1 = 3, T_2^1 = 5, T_3^1 = 11, T_4^1 = 9, T_5^1 = 17, T_6^1 = 19$.

Dabar rasime kritinį kelią. Skaičių μ_0 , kuris lygus kritiniam laikui, gavome sudėję μ_1 – ilgiausio kelio iš P_1 į P_6 ilgį ir t_{01} – darbo (P_0, P_1) trukmę. Vadinasi, darbas (P_0, P_1) yra kritiniame kelyje. Skaičių μ_1 gavome sudėję μ_2 – ilgiausio kelio iš P_2 į P_6 ilgį ir t_{12} – darbo (P_1, P_2) trukmę. (P_1, P_2) yra kitas darbas kritiniame kelyje. Panašiai samprotaudami įsitikiname, kad darbai (P_2, P_3) ir (P_3, P_6) taip pat yra kritiniame kelyje. Taigi kritinis kelias yra $\{(P_0, P_1), (P_1, P_2), (P_2, P_3), (P_3, P_6)\}$.

Laiko rezervai (1)

Analizuojant projekto realizavimą, reikia žinoti įvairius laiko rezervus. Jie apibrėžiami remiantis parametrais T_j^0 ir T_j^1 .

Laisvės intervalas

Laiko tarpas $[T_j^0, T_j^1]$ vadinamas įvykio P_j laisvės intervalu, arba rezerviniu intervalu.

Pilnasis laiko rezervas

Darbo (P_i, P_j) pilnuoju laiko rezervu vadinamas skaičius
$$p_{ij} = T_j^1 - T_j^0 - t_{ij}.$$

Pilnasis laiko rezervas nusako, kiek darbą (P_i, P_j) galima užtęsti (vėliau pradėti) nedidinant kritinio laiko. Išnaudojus pilnąjį (P_i, P_j) laiko rezervą, vėlesni darbai (P_j, P_k) pradedami tik vėliausiuoju laiku T_j^1 .

Nagrinėjamo tinklinio grafiko (p.91) parametrai T_j^0 , T_j^1 bei intervalai $[T_j^0, T_j^1]$

P_j	P_0	P_1	P_2	P_3	P_4	P_5	P_6
T_j^0	0	3	5	11	6	14	19
T_j^1	0	3	5	11	9	17	19
$[T_j^0, T_j^1]$	—	—	—	—	[6, 9]	[14, 17]	—

Laisvasis laiko rezervas

Darbo (P_i, P_j) laisvuoju laiko rezervu vadinamas skaičius

$$l_{ij} = T_j^0 - T_j^0 - t_{ij}.$$

Laisvasis laiko rezervas nurodo, kiek darbą (P_i, P_j) galima užtęsti (vėliau pradėti), netrukdamt anksčiausiuoju laiku T_j^0 pradėti vėlesnių darbų. Naudojantis darbo (P_i, P_j) laisvuoju laiko rezervu, gali sumažėti anksčiau pradėtų darbų (P_k, P_i) pilnieji laiko rezervai, tačiau vėlesniųjų darbų pilnieji laiko rezervai nesikeičia.

Nepriklausomasis laiko rezervas

Darbo (P_i, P_j) nepriklausomuoju laiko rezervu vadinamas skaičius

$$m_{ij} = \max \{0, T_j^0 - T_i^0 - t_{ij}\}$$

Išnaudojus šį rezervą, P_i įvyksta vėliausiuoju laiku T_i^1 , o P_j - anksčiausiuoju laiku T_j^0 . Nepriklausomasis laiko rezervas nepriklauso nuo kitų darbų laiko rezervų panaudojimo.

Užd. Raskite nagrinėjamo tinklinio grafiko (p.91) laiko rezervus

Nagrinėjamo tinklinio grafiko (p.91) laiko rezervai

Darbas	Darbo trukmė	Laiko rezervai		
		p_{ij}	l_{ij}	n_{ij}
(P_0, P_1)	3	0	0	0
(P_0, P_2)	4	1	1	1
(P_1, P_2)	2	0	0	0
(P_1, P_3)	7	1	1	1
(P_1, P_4)	3	3	0	0
(P_1, P_5)	5	9	6	6
(P_2, P_3)	6	0	0	0
(P_2, P_4)	1	3	0	0
(P_3, P_6)	8	0	0	0
(P_4, P_5)	8	3	0	0
(P_5, P_6)	2	3	3	0

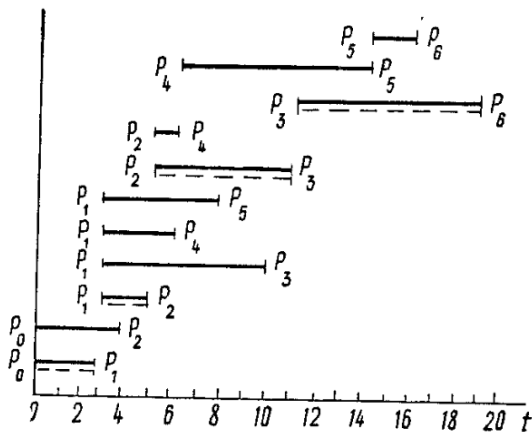
Tiesinė projekto diagrama (1)

Tinklinis grafikas nurodo darbų eilę, tačiau jis nenusako, kokie darbai turi būti vykdomi kiekvienu laiko momentu. Kai projektai nedideli, patogu naudotis vadinamąja tiesine diagrama.

Tiesinė projekto diagrama (2)

Pasirinktu masteliu horizontalioje ašyje atidedame laiką. Kiekvieną darbą vaizduojame laiko ašiai lygiagrečia atkarpa, kurios ilgis lygus darbo trukmei. Įvykio P_0 laiką laikome nuliniu. Atkarpas išdėstome viena virš kitos pagal darbų sąrašą. Atkarpą (P_i, P_j) brėžiame taip, kad jos pradžia P_i būtų vertikalėje, kurioje yra toliausias iš atkarpų (P_k, P_i) dešiniųjų galų. Todėl atkarpos (P_i, P_j) pradžia sutampa su įvykio P_i anksčiausiuoju laiku T_i^0 .

Nagrinėjamo tinklinio grafiko (p.91) tiesinė diagrama



Tiesinė projekto diagrama (3)

Remiantis diagrama, galima rasti tinklinio grafiko parametrus.

Kritinis laikas

Kritinis laikas T_n lygus paskutinio diagramos taško P_6 laikui.

Kritinis kelias

Kritinis kelias brėžinyje pažymėtas punktyrine linija. Jį randame šitaip. Imame darbą (P_3, P_6) , kurio pabaiga sutampa su kritiniu laiku. Šios atkarpos kairysis taškas P_3 pagal laiką sutampa su darbo (P_2, P_3) dešiniuoju tašku P_3 . Atkarpos (P_2, P_3) kairysis taškas sutampa su darbo (P_1, P_2) dešiniuoju tašku P_2 ir t.t.

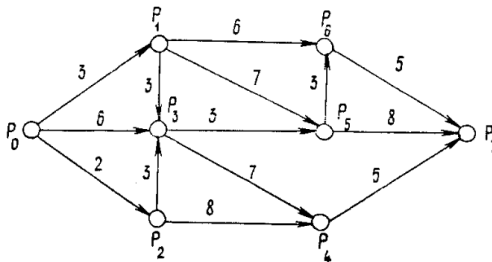
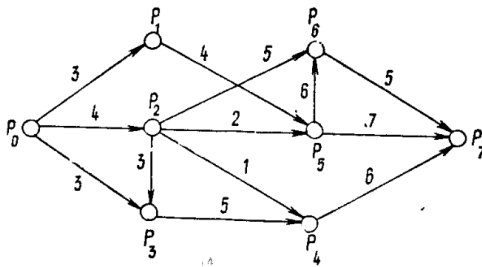
Tiesinė projekto diagrama (4)

Naudojantis diagrama, nesunku rasti laiko rezervus ir paaiškinti jų prasmę. Pilnasis darbo (P_i, P_j) laiko rezervas rodo, kiek galima nukelti atkarpą (P_i, P_j) į dešinę, nekeičiant kritinio laiko. Pavyzdžiui, (P_1, P_5) galima nukelti per 6 laiko vienetų iki (P_5, P_6) , po to dar per 3 laiko vienetų kartu su (P_5, P_6) . Vadinasi, pilnasis darbo (P_1, P_5) laiko rezervas yra $p_{15}=9$. Laisvasis darbo (P_i, P_j) laiko rezervas rodo, kiek galima nukelti atkarpą (P_i, P_j) į dešinę, nekeičiant atkarpų (P_j, P_k) . Nepriklausomasis darbo (P_i, P_j) laiko rezervas rodo, kiek atkarpą (P_i, P_j) galima nukelti į dešinę, jeigu ji jau buvo nukelta pagal ankstesnių darbų pilnuosius laiko rezervus, o vėlesni darbai nekeliami.

Tiesinė projekto diagrama (5)

Jeigu darbų laiko rezervai dideli, tinklinis grafikas „lankstus“, tai darbus galima užtęsti nedidinant viso projekto realizavimo laiko. Jeigu daugelio darbų laiko rezervai lygūs nuliui, tai tinklinis grafikas yra „kietas“. Tada mažai galimybių sutaupyti resursų arba pigiau realizuoti projektą nedidinant realizavimo laiko.

Raskite tinklinių grafikų kritinį kelią, laiko rezervus ir nubraižykite tiesines diagramas



Tinklinio planavimo uždaviniai. Resursų paskirstymas.

Realizuojant projektą, reikalingi įvairūs resursai: darbo jėga, įrengimai, finansai. Kadangi resursai riboti, tai iškyla įvairūs jų paskirstymo uždaviniai. Pagrindiniai uždaviniai yra šitokie:

1. Kaip skirstyti resursus, kad visą darbų kompleksą būtų galima atlikti per trumpiausią laiką?

2. Reikia rasti mažiausią resursų kiekį, reikalingą visiems darbams atlikti per tam tikrą laiką.

Bendrujų metodų tokiems uždaviniams spręsti nėra.

Projekto realizavimo kainos minimizavimas (1)

Paprastai, darbų kainos (iškaudos atliekant darbus) priklauso nuo darbų trukmės. Dažnai trukmei didėjant kaina mažėja. Tačiau būna atvejų, kai didėjant trukmei didėja ir išlaidos.

Projekto realizavimo kainos minimizavimas (2)

Nagrinėsime projektą, kai nustatyta kiekvieno darbo (P_i, P_j) minimali trukmė d_{ij} . Darbo kaina c_{ij} apskaičiuojama pagal formulę

$$c_{ij} = -a_{ij}t_{ij} + b_{ij}, \quad a_{ij} \geq 0, \quad b_{ij} > 0,$$

čia t_{ij} - darbo (P_i, P_j) trukmė. Reikia rasti projekto realizavimo planą, pagal kurį darbų kompleksas būtų atliktas per trumpiausią laiką su mažiausiomis sąnaudomis.

Iš formulės išplaukia, kad, didinant darbo trukmę, jo kaina mažėja. Vadinasi, kiekvieno darbo trukmė turi būti kiek galima didesnė. Kita vertus, trumpiausias laikas lygus kritiniam laikui, imant $t_{ij} = d_{ij}$. Vadinasi, galima užtęsti tik nekritinius darbus iki jie taps kritiniais.

Projekto realizavimo kainos minimizavimas (3)

Tarkime, kad apskaičiuoti parametrai $T_j^0, T_j^1, j=0, 1, \dots, n$, ir rastas kritinis kelias bei kritinis laikas. Nekritinių darbų aibę pažymėkime R , o įvykių, priklausančių kritiniam keliui, aibę – K . Sakykime, T_j – planuojamas įvykio P_j laikas, $j=0, 1, \dots, n$. Kad darbų kompleksas būtų atliktas per kritinį laiką, dydžiai T_j turi tenkinti šias sąlygas:

$$T_j = T_j^0 = T_j^1, \quad \text{kai } P_j \in K; \quad (3.2)$$

$$T_j - T_i = t_{ij} \geq d_{ij}, \quad \text{kai } (P_i, P_j) \in R.$$

Tada projekto realizavimo kaina yra

$$z = \sum_{(P_i, P_j)} (-a_{ij}(T_j - T_i) + b_{ij}). \quad (3.3)$$

Uždavinio matematinis modelis (1)

Reikia rasti T_j , $j = 0, 1, \dots, n$ reikšmes, tenkinančias (3.2) sąlygas, su kuriosmis (3.3) funkcija įgyja mažiausią reikšmę:

$$\min \sum_{(P_i, P_j)} (-a_{ij}(T_j - T_i) + b_{ij}), \quad (3.4)$$

$$\begin{aligned} T_j - T_i &\geq d_{ij}, & \text{jei } (P_i, P_j) \in R; \\ T_i &= T_i^0, & \text{jei } P_i \in K. \end{aligned} \quad (3.5)$$

Uždavinio matematinis modelis (2)

Tarkime, kad reikia rasti projekto realizavimo planą, pagal kurį darbų kompleksas būtų atliktas su mažiausiomis sąnaudomis per tam tikrą laiką T , kuris yra didesnis už kritinį. Matematinis modelis bus šitoks: reikia rasti

$$\min \sum_{(P_i, P_j)} (-a_{ij}(T_j - T_i) + b_{ij}), \quad (3.6)$$

kai

$$\begin{aligned} T_j - T_i &\geq d_{ij} \text{ visiems } (P_i, P_j), \\ T_0 &= 0, \quad T_n \leq T. \end{aligned} \quad (3.7)$$

Pažymėsime, kad čia T_n ne kritinis, o planuojamas įvykio P_n laikas.

Uždavinio matematinis modelis (3)

Jeigu darbų kainos apskaičiuojamos remiantis netiesinėmis funkcijomis, tai sudarę matematinį modelį gauname netiesinio programavimo uždavinį. Pavyzdžiui, kai

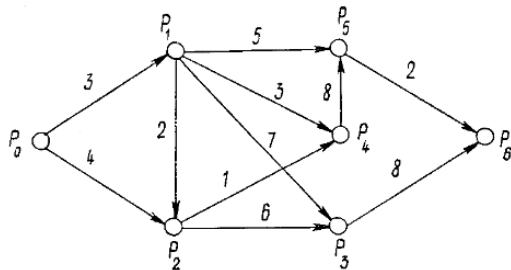
$$c_{ij} = \frac{a_{ij}}{t_{ij}}, \quad a_{ij} > 0,$$

turime iškiląjo programavimo uždavinį su tikslo funkcija

$$\sum_{(P_i, P_j)} \frac{a_{ij}}{T_j - T_i}$$

ir tiesiniais (3.5) arba (3.7) apribojimais.

Pavyzdys. Nagrinėjime projektą



Pavyzdys (1)

Tarkime, kad skaičiai, surašyti prie grafiko briaunų, reiškia darbų (P_i, P_j) minimalią trukmę d_{ij} . Darbų kainos c_{ij} apskaičiuojamos pagal formules

$$c_{01} = -5t_{01} + 100,$$

$$c_{23} = -4t_{23} + 80,$$

$$c_{02} = -3t_{02} + 100,$$

$$c_{24} = -5t_{24} + 120,$$

$$c_{12} = -10t_{12} + 50,$$

$$c_{36} = -3t_{36} + 100,$$

$$c_{13} = -8t_{13} + 150,$$

$$c_{45} = -10t_{45} + 250,$$

$$c_{14} = -2t_{14} + 60,$$

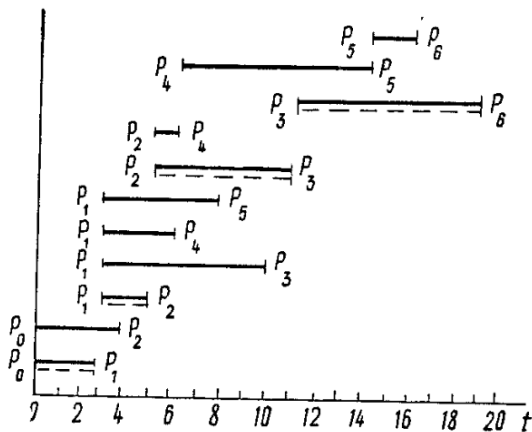
$$c_{60} = -4t_{60} + 40.$$

$$c_{15} = -10t_{15} + 100$$

Nagrinėjamo tinklinio grafiko parametrai T_j^0 ir T_j^1

P_j	P_0	P_1	P_2	P_3	P_4	P_5	P_6
T_j^0	0	3	5	11	6	14	19
T_j^1	0	3	5	11	9	17	19
$[T_j^0, T_j^1]$	—	—	—	—	[6, 9]	[14, 17]	—

Nagrinėjamo tinklinio grafiko tiesinė diagrama



Reikia rasti projekto realizavimo planą, pagal kurį darbų kompleksas būtų atliktas per trumpiausią laiką su mažiausiomis sąnaudomis.

Sudarysime matematinį modelį. Parametrai T_j^0 ir T_j^1 apskaičiuoti (2.1 lentelė). Kritinis kelias yra $\{(P_0, P_1), (P_1, P_2), (P_2, P_3), (P_3, P_6)\}$. Kritinis (trumpiausias) laikas – $T_6 = 19$. Nekritinių darbų aibė – $R = \{(P_0, P_2), (P_1, P_3), (P_1, P_4), (P_1, P_5), (P_2, P_4), (P_4, P_5), (P_5, P_6)\}$. Įvykių, priklausančių kritiniam keliui, aibė $K = \{P_0, P_1, P_2, P_3, P_6\}$. Pagal (3.4), (3.5) užrašome tiesinio programavimo uždavinį: reikia rasti

$$\min (8T_0 + 25T_1 - 4T_2 - 9T_3 + 3T_4 - 16T_5 - 7T_6 + 1150),$$

kai

$$T_2 - T_0 \geq 4, \quad T_4 - T_2 \geq 1,$$

$$T_3 - T_1 \geq 7, \quad T_5 - T_4 \geq 8,$$

$$T_4 - T_1 \geq 3, \quad T_6 - T_5 \geq 2,$$

$$T_5 - T_1 \geq 5,$$

$$T_0 = 0, \quad T_1 = 3, \quad T_2 = 5, \quad T_3 = 11, \quad T_6 = 19.$$

Eliminavę dydžius T_j , turinčius tik vieną reikšmę, gauname šitoki uždavinį: reikia rasti

$$\min (3T_4 - 16T_5 + 973),$$

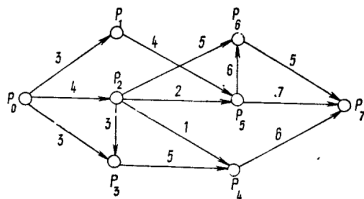
kai

$$T_5 - T_4 \geq 8,$$

$$T_4 \geq 6, \quad T_8 \geq 8, \quad T_5 \leq 17.$$

Aišku, kad šio uždavinio sprendinys yra $T_4^* = 6, T_5^* = 17$. Vadinasi, įvykiams $P_i, i = 0, 1, \dots, 6$, reikia planuoti šitoki laiką: $T_0^* = 0, T_1^* = 3, T_2^* = 5, T_3^* = 11, T_4^* = 6, T_5^* = 17, T_6^* = 19$. Optimali darbo (P_i, P_j) trukmė yra $t_{ij}^* = T_j^* - T_i^*$. Taigi $t_{01}^* = 3, t_{02}^* = 5, t_{12}^* = 2, t_{13}^* = 8, t_{14}^* = 3, t_{15}^* = 14, t_{23}^* = 6, t_{24}^* = 1, t_{36}^* = 8, t_{45}^* = 11, t_{56}^* = 2; z_{\min} = 719$.

Uždaviniai (1)



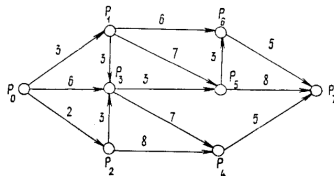
60 pav.

3. Skaičiai, surašyti prie grafiko briaunų (60 pav.), reiškia darbų (P_i, P_j) minimalią trukmę. Darbų kainos apskaičiuojamos pagal šias formules:

$$\begin{aligned} c_{01} &= -10t_{01} + 80, & c_{25} &= -4t_{25} + 140, \\ c_{02} &= -3t_{02} + 100, & c_{26} &= -2t_{26} + 60, \\ c_{03} &= -4t_{03} + 70, & c_{34} &= -3t_{34} + 50, \\ c_{15} &= -3t_{15} + 60, & c_{47} &= -2t_{47} + 80, \\ c_{23} &= -5t_{23} + 100, & c_{56} &= -5t_{56} + 100, \\ c_{34} &= -t_{34} + 30, & c_{57} &= -3t_{57} + 70, \\ & & c_{67} &= -2t_{67} + 100. \end{aligned}$$

Raskite projekto realizavimo planą, pagal kurį darbų kompleksas būtų atliktas per trumpiausią laiką su mažiausiomis sąnaudomis.

Uždaviniai (2)



61 pav.

4. Skaičiai, surašyti prie grafiko briaunų (61 pav.), reiškia darbų (P_i, P_j) minimalią trukmę. Darbų kainos apskaičiuojamos pagal šias formules:

$$c_{01} = -10t_{01} + 180,$$

$$c_{24} = -4t_{24} + 140,$$

$$c_{02} = -3t_{02} + 60,$$

$$c_{34} = -2t_{34} + 100,$$

$$c_{03} = -5t_{03} + 150,$$

$$c_{35} = -5t_{35} + 120,$$

$$c_{13} = -2t_{13} + 70,$$

$$c_{47} = -8t_{47} + 150,$$

$$c_{15} = -t_{15} + 80,$$

$$c_{56} = -5t_{56} + 70,$$

$$c_{16} = -2t_{16} + 70,$$

$$c_{57} = -t_{57} + 140,$$

$$c_{23} = -6t_{23} + 150,$$

$$c_{67} = -8t_{67} + 200.$$

Raskite projekto realizavimo planą, pagal kurį darbų kompleksas būtų atliktas su mažiausiomis sąnaudomis per laiką $T=25$.