ҚАЗАҚСТАН РЕСПУБЛИКАСЫ
БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Қ. И. Сәтбаев атындағы
Қазақ ұлттық техникалық зерттеу университеті

Л. Ш. Балғабаева

ОПЕРАЦИЯЛАРДЫ ЗЕРТТЕУ НЕГІЗДЕРІ

Қазақстан Республикасы Білім және ғылым министрлігі
оқулық ретінде ұсынған

Алматы 2016

ӘОЖ 519.81(075)
КБЖ 22.18 я 7
Б 22

Пікір жазғандар:
Түкеев У.Ә., техн. ғыл. докторы, профессор, ҚазҰУ;
Өтепбергенов И.Т., техн. ғыл. докторы, профессор, АЭжБУ;
Өскенбаева Р.К., техн. ғыл. докторы, профессор, Халықаралық ақпараттық технологиялар университеті;
Қалижанова Ә.Ұ., физ.-мат. ғыл. канд., доцент, ҚазҰТУ.

Қазақстан Республикасы Білім және ғылым министрлігінің 2016 жылғы басылым жоспары бойынша басылды.

Балғабаева Л.Ш.
Б 22 Операцияларды зерттеу негіздері: Оқулық / Л.Ш. Балғабаева – Алматы: ҚазҰТЗУ, 2016. –269 б. Сурет – 29. Кесте -120. Библиогр. – 42 атау.
ISBN 978-601-228-940-4

Оқулық «Операцияларды зерттеу негіздері» курсының бағдарламасына сәйкес жазылған. Оқулықта математикалық бағдарламалаудың негізгі теориялық және есептік аспектілері келтірілген. Әрбір бөлімнің алғашқы параграфтары бөлімде қолданылатын анықтамалар мен ұғымдарға арналған. Есептеу алгоритмдеріне айрықша мән берілген. Есептерді шешу әдістерінің алгоритмдері келтірілген, оған қоса типтік есептерді шешу жолдары егжей-тегжейді көрсетілген. Көптеген есептер есепті шешудің барлық кезеңдерінде қарастырылған – есептің қойылымынан бастап, табылған нәтижені талдауға дейін. Әрбір бөлімнің соңында оқып үйренген материалды пысықтауға арналған сұрақтар мен жаттығулар келтірілген.

Оқулық 5В070300 – «Ақпараттық жүйелер» және 5В070400 – «Есептеу техникасы және бағдарламалық қамтама» мамандықтары бойынша білім алатын бакалаврларға, техникалық және экономикалық кибернетика бойынша мамандандырылған университеттердің студенттеріне арналған; оқулықты технологиялық және экономикалық процестерді оңтайландырудың математикалық әдістерін оқып зерттеп жүрген магистранттар да қолдана алады. Докоранттарға, ғылыми қызметкерлерге және жоспарлау мен басқарудың математикалық әдістеріне қызығушылық танытып жүрген тұлғаларға өз пайдасын тигізеді.

ӘОЖ 519.81(075)
КБЖ 22.18 я 7

ISBN 978-601-228-940-4

© Балғабаева Л.Ш., 2016
© ҚазҰТЗУ, 2016

АЛҒЫ СӨЗ

Оқулық Қ.И.Сәтбаев атындағы ҚазҰТЗУ-де «Ақпараттық жүйелер» мамандығы бойынша бакалавриат студенттеріне оқылған «Операцияларды зерттеу негіздері» пәнінің дәрістеріне негізделген. Бұл дәрістер берілген мамандыққа арналған силлабустар мен оқу-әдістемелік кешендерде көрініс тапты, сондай-ақ бірқатар өзгерістерге ұшырап, көптеген көрнекі сипаттағы мысалдармен толықтырылды.

Бірінші тарауда алға қойылған мәселеге арналған жүйелік тәсіл сипатталды.

Одан кейінгі екі тарау сызықтық бағдарламалаудың негізгі класикалық әдістері мен есептеріне арналды. Сипатталған әдістердің есептеу аппараты ретінде қарапайым және түрлендірілген жордан шығарулары қолданылды.

Транспорт есебі сызықтық есеп болып табылады, оны сызықтық бағдарламалау әдістерімен шешуге болады. Алайда оны шешу үшін симплекс әдісін қолдану тым қолайсыз болады (есептің өлшемінің үлкендігі себебінен). Төртінші тарауда транспорт есебін шешудің ең кең таралған әдісі – потенциалдар әдісі келтірілді, ол – осы есеп шарттарының ерекшеліктерін қолданатын және симплекс әдісімен салыстырғанда мақсатқа айтарлықтай жылдам әрі оңай жетуге мүмкіндік беретін әдіс.

Бесінші тарау бүтін сандық шешімдері бар сызықтық бағдарламалау есептеріне арналды. Мұндай есептердің ерекше болуына байланысты оларды шешу үшін шектеулі қадамнан кейін бүтін сандық шешімге алып келетін арнайы әдістер талап етіледі.

Оқулықтың одан кейінгі үш тарауында динамикалық, тораптық бағдарламалау әдістері және коммивояжер есептері ұғынықты түрде сипатталды.

Бейсызықтық бағдарламалау есебін шешу үшін бүгінгі таңда күрделілігі жағынан әр түрлі болып келетін көптеген

әдістер ұсынылды. Бірақ бұл әдістердің ешқайсысы әмбебап емес әрі бейсызықтық есептердің барлығына бірдей жарамды емес. Әрбір әдістің өзіндік артықшылықтары мен кемшіліктері бар. Бейсызықтық бағдарламалау есептеріне арналған тарауда есептің математикалық моделіне тәуелді әдістер таңдалды. Келтірілген әдістердің барлығы сандық мысалдармен көрсетілді.

Оныншы тарау ойындар және статистикалық шешімдер теориясына және оларды сызықтық бағдарламалау есебіне келтіруге арналды.

Осы кітап «Операцияларды зерттеу негіздері», «Шешімдер қабылдау теориясы», «Оңтайландыру модельдері мен әдістері» курстарын оқитын техникалық мамандық студенттеріне, ақпараттық жүйелерді жобалауға қатысатын магистранттарға, инженерлерге арналған.

Теория бойынша, сондай-ақ өздігінен шешуге арналған есептер бойынша қосымша мәліметтерді әдебиет тізімінде келтірілген кітаптардан алуға болады.

Автор қолжазбаның жақсаруына себеп болған құнды ескертулері мен тілектері үшін Қ.И. Сәтбаев атындағы ҚазҰТЗУ-ң «Ақпараттық технологиялар» кафедрасының қызметкерлеріне шын ризашылығын білдіреді.

Автор

КІРІСПЕ

Күрделі процестерді, құбылыстарды зерттеу кезінде, жаңа ғимараттарды құрастыру кезінде және т.б. жағдайларда модельдеу қолданылады. Модельдеудің алғашқы тәсілдерінің бірі ұқсастық әдісі болып табылады. Модельдеудің бұл әдісінде зерттелетін құбылыс эксперименталды ортада, яғни басқа «кіші» масштабта жүзеге асырылады. Осылайша жұмыс істеп тұрған модельде белгілі бір құбылыс зерттеледі.

Модель және модельдеу дегеніміз не?

Модель – бұл арнайы құрылатын объект, мұнда зерттелетін объектіге зерттеу жүргізу мақсатында оның айқын сипаттамалары жүзеге асырылады, ал модельдеу дегеніміз – зерттелетін объектінің қарастырылып отырған сипаттамаларының нақты айқын көрінісі. Модельдеу ғылыми абстракцияның маңызды құралы болып табылады, ол объектінің зерттеуге қажетті қасиеттері, өзара байланыстары, құрылымдық, функционалды параметрлері сияқты сипаттамаларын бөліп алуға, ерекшелеуге және талдауға мүмкіндік береді.

Модель мен модельдеу объектісінің сәйкес келу дәрежесі әр түрлі болуы мүмкін. Мысалы, функционалды модельдер объектінің тек жүріс-тұрысы мен қызметін ғана көрсетеді. Олар кибернетикада «қара жәшік» әдісімен модельдеу үшін кеңінен қолданылады. «Қара жәшік» деп ішкі құрылысы бақылаушыға белгісіз және ол үшін маңызды емес болып табылатын жүйені айтамыз, бірақ бақылаушы осы жүйенің «кірістері» мен «шығыстарын» зерттей алады.

Модельдер қандай да бір физикалық объектілердің (физикалық модель, мысалы, кішірейтілген объектілер, жаттығу құрылғылары) көмегімен, сонымен қатар абстрактілі объектілердің (қандай да бір көрсетілген абстрактілі модель) көмегімен де жүзеге асырылуы мүмкін. Мұндай абстрактілі

модель модельдеу объектісінің сипаттамасын беретін математикалық өрнектер болуы мүмкін (математикалық модель). Көбінесе математикалық модель деп тек сандық сипаттамаларды қамтитын, формула түрінде жазылған модельді айтады. Нақты сандық сипаттамалары бар модельдерді сандық модельдер деп атайды; логикалық өрнектердің көмегімен жазылған модельдер – логикалық (алгоритмдердің блок-схемалары), ал графикалық бейнелермен жазылған модельдер – графикалық (графиктер, диаграммалар) деп аталады. ЭЕМ көмегімен жүзеге асырылған модельдер машиналық (электронды) модельдер деп те аталады. Модельдеу жүзеге асырылатын ЭЕМ-ң типіне байланысты аналогтық және дискретті (цифрлық) модельдер болып бөлінеді. Осы модельдердің барлығының негізінде математикалық модель жатыр.

Математикалық модельдеу модельдеудің ең жетілген әрі тиімді әдісі болып табылады. Дәл осы модельдеу жолы математикалық талдаудың күшті құралдарын қолдануға мүмкіндік береді, себебі математикалық модельдер тікелей ақиқатпен жанаса алмайды, олар тек қандай да бір құбылыстар шеңберінің математикалық модельдерімен ғана жанаса алады. Егер модель шынайы жағдайды толықтай дерлік көрсететін болса, мұндай модельдің зерттеу нәтижелері практикалық мүддеге ие болады. Математикалық модельдердің кемелдік дәрежесі мен қолданылатын математикалық аппарат ғылымның даму деңгейін сипаттайды.

Халықтық-шаруашылық есептерді зерттеп шешу үшін математикалық аппаратты кеңінен қолданған сайын жаңа математикалық әдістерді құру қажеттілігі туындады. Осының арқасында сызықтық бағдарламалау, бүтін сандық бағдарламалау, динамикалық бағдарламалау, графтар теориясы, ойындар теориясы және т.б. сияқты бірқатар жаңа математикалық пәндер пайда болды. Осы қолданбалы математикалық пәндердің зерттеу пәні халық шаруашылығының белгілі бір процестерінен туындаған және солармен байланысқан математикалық модельдер болып табылады (өнімді алу тәсілдерін таңдау, технологияны таңдау,

жабдықтарды қолдану, процесті ұйымдастыру). Шаруашылық жоспарды қарастырылып отырған жүйенің әр түрлі деңгейлерінде құрастыру іс жүзінде маңызды мәселе болып табылады. Құрылған жоспардың сапасы олардың негізіне салынған шешімдерге байланысты. Осы шешімдерді дұрыс таңдаған жағдайда бірдей шығын көлемімен әлдеқайда үлкен экономикалық нәтижеге қол жеткізуге болады.

Бірнеше мүмкін шешімдердің арасынан ең тиімдісін, яғни оңтайлысын таңдау маңызды мәселе болып табылады. Сондықтан өндіріс процестерін жоспарлау кезінде ең тиімді, оңтайлы шешімді табу әдістері үлкен маңызға ие болады, бұл әдістер математикалық (оңтайлы) бағдарламалау деген атаумен біріктіріледі.

1. ЖҮЙЕЛІК ТАЛДАУ ЖӘНЕ ОПЕРАЦИЯЛАРДЫ ЗЕРТТЕУ ӘДІСТЕМЕСІ

1.1. Математикалық бағдарламалау. Даму тарихы

Материалды математикалық формалдауды қолдана отырып жеке экономикалық проблемаларды математикалық зерттеу сонау XIX ғасырда жүргізілген. Кеңейтілген ұдайы өндіріс процесін математикалық талдау кезінде көбінесе алгебралық арақатынастар қолданылды. Экономикалық жүйе динамикасын жалпылама талдау кезінде дифференциалды теңдеулер де қолданылады. Сол кезде XX ғасырға тән жоғары дамыған шаруашылық қажеттіліктері бұдан да нақтырақ экономикалық көрсеткіштер мен сипаттамаларды қажет етті. Бұл КСРО-да XX ғасырдың 20-жылдарында сала аралық баланс жүйесінің құрылуына алып келді. Ол экономикалық жүйелер мен жағдайлардың математикалық модельдерінің дамуы мен зерттелуіне айтарлықтай түрткі болды. КСРО-дағы сала аралық баланс өнімдері американдық экономист В.В. Леонтьевтің жұмысына әсер етті (алғашқы басылымы 1936 ж.). Леонтьев құрған АҚШ өндірісінің және өнімдерін үлестірудің сала аралық моделі әдебиетке «шығындар – өнім» экономикасын талдау әдісі деген атпен енді.

1938 жылы кеңес одағының математигі Л.В. Канторович ғылыми кеңес тәртібінде лушильді станоктар тобын жүктеудің ең тиімді өндірістік бағдарламасын таңдаудың практикалық есебін зерттеуге кірісті. Л.В. Канторович: «Осы максимумға бағытталған және сызықтық теңсіздіктер жүйесімен сипатталатын шектеулері бар есеп тым өзгеше екендігі және класикалық талдаудың белгілі құралдарымен шешілмейтіндігі анықталды. Сол кезде белгілі болғаны – бұл есеп кездейсоқ емес, оңашаланған емес, ол әлі зерттелмеген жаңа есептер класының әдеттегі өкілі болып табылатындығы айқындалды, бұл есептерге экономикалық талдауға, әсіресе әлеуметтік экономикаға тән ең тиімді өндірістік жоспарды табудың түрлі мәселелері келтіріледі. Осы есептер шеңберін және оларды шешу әдістерін оқып үйрену нәтижесінде сызықтық

бағдарламалау деп аталған жаңа ғылыми пән пайда болды». [8] Л.В. Канторовичтің «Өндірісті ұйымдастырудың және жоспарлаудың математикалық әдістері» жұмысының (1939 ж.) пайда болуы экономикалық-математикалық әдістердің дамуында жаңа кезеңге жол ашты. Алайда сызықтық бағдарламалау әдістері ұзақ уақыт бойы өңделмеді және практикаға да енгізілмеді.

Арада 10 жыл өткен соң американдық математик Дж. Данциг берілген есептер класын шешудің тиімді әдісін ойлап тапты, ол – сипмлекс әдісі. «Сызықтық бағдарламалау» термині алғаш рет 1951 жылы Дж. Данциг пен Т. Купманстың еңбектерінде пайда болды. «Бағдарламалау» сөзі табылуы тиіс айнымалылар жиыны әдетте нақты экономикалық объектінің жұмыс бағдарламасын (жоспарын) анықтайтынын білдіреді [1].

Сызықтық бағдарламалау 1955-1965 жылдары КСРО мен АҚШ-та өте қарқынды түрде дамыды, ол кезде ол қолданбалы математиканың ең «танымал» бөлімдерінің бірі болды.

Сызықтық бағдарламалау мен сала аралық баланс «сызықтық экономиканы» сипаттай алады, яғни шаруашылық элементтерінің өзара сызықтық байланыстарын сипаттайды. Алайда тереңірек зерттеу кезінде сызықтық емес сипаттағы байланыстар көрінеді, ол кезде бір элементті өзгерту кезінде басқалары алғашқысына пропорционалды емес түрде өзгереді. Сондықтан сызықтық бағдарламалау әдістерін өңдеуден кейін математикалық бағдарламалаудың басқа бағыттарына да зерттеулер жүргізіле бастады: бейсызықтық, бүтін сандық, стохастикалық және т.б.

КСРО-дағы экономикалық-математикалық және экономикалық-кибернетикалық зерттеулер бірқатар даму кезеңдерінен өтті. Бірінші кезеңде (өткен ғасырдың 50-жылдарының соңы) сала аралық баланстың баланстық модельдері мен эксперименталды экономикалық есептеулер үшін қолданылатын сызықтық бағдарламалаудың жеке модельдері жасалды.

Екінші кезеңде (өткен ғасырдың 60-жылдарының басы) зерттеушілер түрлі типті оңтайландыру модельдерінің жасалуына және олардың жоспарлау есептерін шешуге

практикалық тұрғыда қолданылуына көңіл бөлді. Бұл кезеңде көптеген экономикалық-математикалық модельдер құрылды, олардың негізінде шынайы оңтайлы жоспарларды құру бойынша есептеулер жүргізілді (тасымалдаулардың оңтайлы жоспарлары, транспорттың жылжымалы құрамын пайдаланудың, кәсіпорындардың жабдықтарын жүктеудің оңтайлы жоспарлары, өндірістің жеке салаларын оңтайлы орналастыру, капитал жұмсауды оңтайлы жоспарлау және тарату және т.б.), осылар халық шаруашылығына үлкен табыс алып келді.

Үшінші кезеңде жүйелік тәсіл теориялық зерттеулерде де, математикалық модельдердің практикалық қолданылуында да дамыды. Жеке оңтайландыру модельдерін жасаудан тұжырымдарды қалыптастыруға және халық шаруашылығының оңтайлы қызмет ету теориясын құруға көшті. Жоспарлауға және басқаруға математикалық әдістердің қолданылуы автоматтандырылған басқару жүйелерінің (АБЖ) шегінде жүзеге асырылады, олар халық шаруашылығының түрлі бөлімдері үшін жасалады: кәсіпорындар, салалар, аймақтық-өндірістік кешендер. Халық шаруашылығындағы АБЖ-ларды құру бойынша жұмыс мемлекеттік деңгейде ұйымдастырылған, олар шаруашылық жоспарлар мен олардың жүзеге асырылуының оңтайландырылуын қамтамасыз етеді.

Математикалық бағдарламалау – шектеулері бар көп өлшемді экстремалды есептерді, яғни айнымалылардың өзгеру облысына қойылған шектеулері бар көптеген айнымалылар функциясының экстремумының есептерін шешудің теориясы мен сандық әдістерін жасайтын математика саласы. Математикалық талдаудың бір бөлігі болып табылатын экстремалды есептердің класикалық теориясына қарағанда, айнымалылардың өзгеру облысына шектеулер қойылатын есептерге айрықша көңіл бөлінеді.

Математикалық бағдарламалау өз атауын өзінің құрамындағы сызықтық бағдарламалаудан алған, мұнда айнымалылардың өзгеру облысының шекарасында экстремумды табу ерекше қиындық тудырды. Сызықтық бағдарламалаудағы жетістіктерден кейін ғана дөңес функцияларды дөңес жиында

минималдау есептерін шешудің жаңа тәсілдері (дөңес бағдарламалау) пайда болды және жалпы түрдегі бейсызықтық экстремалды есептерге (бейсызықтық бағдарламалау) қызығушылық артты. Оңтайландыру есептерінің күрделі класын дискретті есептер құрайды, олардың айнымалыларының өзгеру облысы жеке нүктелерден құралады. Осындай есептердің көпшілігі сызықтық бағдарламалаудың қарқынды дамуы кезеңінде құрастырылған. Алайда тек сызықтық бағдарламалау есептеріне жақын есептер ғана өз шешімдерін тапты. Одан ары қарай дискретті бағадрламалаудың сандық әдістері жасалды. Қазіргі уақытта дискретті бағдарламалау – математикалық бағдарламалаудың ең қарқынды дамып жатқан бөлімдерінің бірі. Сызықтық емес дискретті есептердің көпшілігін рекурренттік тәсіл идеясын қолдана отырып шешуге болады. Мұндай әдістер динамикалық бағдарламалау деген атаумен біріктіріледі. Функцияларында шектеулерді анықтайтын кездейсоқ параметрлері бар оңтайландыру есептері стохастикалық бағдарламалау пәнін құрайды.

Математикалық бағдарламалауға айнымалыларының саны шектеусіз экстремалды есептерді шешу әдістері де жатады – ол шексіз өлшемді бағдарламалау.

1.2. Жүйелік талдау және оңтайландыру

Жүйелік талдау – қоғамның мақсатқа бағытталған қызметінің барлық салаларында туындайтын әр түрлі проблемаларды жүйелік тәсіл негізінде және зерттеу объектісін жүйе түрінде көрсету негізінде шешудің айқындалған ғылыми әдістері мен практикалық тәсілдерінің жиынтығы. Жүйелік талдауға тән сипат – проблеманың ең жақсы шешімін іздеу мынадан басталады: жұмыс істеуі кезінде берілген проблема пайда болған жүйе қызметінің мақсаттарын анықтау және реттеу. Сондай-ақ осы мақсаттардың, туындаған проблеманы шешудің мүмкін жолдарының және осыған қажетті ресурстардың арасындағы сәйкестік анықталады.

Жүйелік талдаудың мақсаты – жұмсалған ресурстарды алынған тиімділікпен сандық және сапалық салыстыру тұрғысынан әр түрлі әрекеттердің нұсқаларын толық әрі жан-жақты тексеру болып табылады.

Жүйелік талдау, бірінші кезекте, әлсіз құрылымдалған проблемаларды, яғни элементтері мен байланыстарының құрамы жартылай ғана айқындалған проблемаларды, белгісіздік факторы бар жағдайларда туындайтын және ресімделмейтін элементтері бар, математика тіліне аударылмайтын элементтері бар есептерді шешуге арналған.

Жүйелік талдау шешім қабылдаушы тұлғаға әрекеттердің мүмкін нұсқаларын бағалауға қатаң қарауға және шешімді дайындаушы мамандарға белгісіз болуы мүмкін ресімделмейтін факторлары мен моменттерін ескере отырып, олардың арасынан ең жақсысын таңдауға мүмкіндік береді.

Жүйелік талдау АҚШ-та пайда болды. АҚШ-та жүйелік талдау көптеген мемлекеттік ұйымдарда оқытылды. Ол қорғаныс саласында және ғарыш кеңістігін зерттеу саласында ең құнды қосалқы жетістік болып саналды. АҚШ конгрессінің қос палатасында да өткен ғасырдың 60-жылдары жүйелік талдау мен жүйетехниканы ел проблемаларын шешу үшін адам ресурстарын толығымен пайдалану мақсатында қолдану туралы заң жобалары ендірілді.

Сондай-ақ, жүйелік талдауды ірі өндірістік кәсіпорындардағы басшылар мен инженерлер қолданды. Жүйелік талдау әдістерін өндірісте және сауда саласында қолданудың мақсаты – көп пайда табудың жолдарын іздеу.

АҚШ-та жүйелік талдау әдістерін қолдану мысалы ретінде бағдарламалық жоспарлау жүйесін айтуға болады, ол «жоспарлау – бағдарламалау – бюджетті құру» (ЖББ) деген атпен де танымал, немесе қысқаша түрде – «бағдарламалық қаржыландыру».

ЖББ жүйесінен басқа АҚШ-та жүйелік талдаудың негізінде жұмыс істейтін бірқатар болжау және жоспарлау жүйелері қолданылады. Атап айтқанда, НИОКР-ды болжап жоспарлау үшін «ПАТТЕРН» ақпараттық жүйесі қолданылатын; «Аполлон» ғарыштық жобасын жасаудың барлық кезеңдерінде

жетекшілік ету үшін «ФЕЙМ» автоматтандырылған ақпараттық жүйесі қолданылды; «КВЕСТ» және «СКОР» жүйелерінің көмегімен әскери тапсырмалар мен оларды жүзеге асыруға қажетті ғылыми-техникалық құралдар арасындағы сандық байланыс орнатылды.

Осы жүйелердің ең басты әдістемелік ерекшелігі «мақсаттар бұтағын» құру мақсатында әрбір проблеманы төмен деңгейлі бірнеше есептерге біртіндеп бөлшектеу принципі болып табылады.

Қарастырылып отырған жүйелер ғылыми және техникалық проблемалардың шешілу мерзімдерін және жұмыстардың өзара тиімділігін анықтауға мүмкіндік берді, шешім қабылдау кезінде тар ведомстволық тәсілден өту арқылы, интуициялық және ерікті шешімдерден, сондай-ақ белгіленген мерзімде орындалуы мүмкін емес жұмыстардан бас тарту арқылы қабылданатын шешімдердің сапасының артуына ат салысты.

Сонымен қатар, АҚШ-тағы соңғы онжылдықтағы басқару тәжірибесі көрсеткендей, «жүйелік талдау» термині бұрынғыдай жиі қолданылмайды. Күрделі шешімдерді негіздеуге арналған көптеген тәсілдер басқа – «бағдарламалық талдау», «саясатты талдау», «нәтижелерді талдау» және т.с.с. атаулармен қолданылып дамып келді. Аталған талдау түрлерінің «жаңалығы» олардың атауында ғана болып табылады. Олардың әдіснамалық және әдістемелік негізі жүйелік талдау, жүйелік тәсіл идеологиясы болып қала береді.

Жүйелік талдау – шешімдер қабылдауға арналған ғылыми, жан-жақты тәсіл. Проблема толықтай қарастырылып зерттеледі, басқару объектісінің даму мақсаттары мен оларды жүзеге асырудың түрлі жолдары анықталады. Бұл жерде ортақ мақсатқа бағыттау үшін басқару объектісінің түрлі бөліктерінің, жеке орындаушылардың жұмыстарын үйлестіру қажеттілігі туындайды.

Ешқандай ғылым бір күнде құрыла салмайды, ол белгілі бір есептер класына деген қызығушылықтың артуы мен осы есептерді шешуге болатын ғылыми принциптердің, әдістер мен құралдардың даму деңгейі сәйкес келуі нәтижесінде пайда болады. Жүйелік талдау да осылай пайда болған. Оның тарихы

тереңде жатыр. Алғашқы адамның өзі өзіне баспана салу үшін түйсіксіз-ақ жүйелі түрде ойланған. Жүйелік талдау Екінші дүниежүзілік соғыс кезінде ғылыми пән ретінде тіркелді, бастапқыда әскери есептерге қолданылды, ал соғыс аяқталған соң азаматтық қызметтің түрлі салаларының есептеріне қолданылды, ол онда кең ауқымды практикалық есептерді шешудің тиімді құралына айналды.

Дәл осы уақытта жүйелік талдаудың жалпы негіздерінің дамығаны соншалықты, оларды білімнің өзіндік тармағы ретінде ресімдей бастады. Жүйелік талдау әдістерінің құрылуының арқасында адам қызметінің барлық салаларындағы басқару қолөнер немесе таза өнер деңгейінен (бұлар көбінесе жеке адамдардың қабілеті мен олардың жинақтаған тәжірибесіне тәуелді болатын) ғылым деңгейіне дейін көтерілді деп толық сеніммен айтуға болады.

Біздің заманда білім жоғары қарқынмен ілгерілеп дамып келеді, ол, бір жағынан, өмірдің түрлі салаларынан жаңа фактілерді, мәліметтерді ашып жинақтауға алып келді де, адамзатқа оларды жүйелендіруге, дербестен ортақты, өзгермеліден тұрақтыны табуға мәжбүр етті. Екінші жағынан, білімнің артуы оны игеруде қиындықтар тудырады, ғылым мен тәжірибеде қолданылатын бірқатар әдістердің тиімсіздігін айқындайды. Оның үстіне, қалыптасып қалған ұғымдар мен пайымдаулардан құралған әлемнен айтарлықтай ерекшеленетін Жер жүзі мен субатомдық әлемнің тереңіне бойлау нәтижесінде жеке ғалымдардың санасында материяның пайда болу және даму заңдарының жалпы негізділігіне күмән туды. Соңында, түрлендіруші қызмет түріне ие болып келе жатқан таным процесінің өзі адамның табиғат дамуындағы субъект ретіндегі рөлі туралы, адам мен табиғаттың өзара қарым-қатынас жасауы туралы, және осыған сәйкес табиғаттың даму заңдары мен олардың әрекетінің жаңадан түсінілуін қалыптастыру туралы сұрақты шиеленістіреді. Мәселе мынада: адамның түрлендіруші қызметі табиғи жүйелердің даму жағдайларын өзгертеді де, қозғалыстың жаңа заңдарының, тенденцияларының пайда болуына ықпал етеді. Әдіснама саласындағы зерттеулер қатарында жүйелік тәсіл және «жүйелік қозғалыс» айрықша

орынға ие. Жүйелік қозғалыстың өзі өзгеріске ұшырап, түрлі бағыттарға бөлінді: жалпы жүйелер теориясы, жүйелік тәсіл, жүйелік талдау, әлемнің жүйелілігінің философиялық мағынасы. Жүйелік зерттеу әдіснамасының ішінде бірқатар аспектілер бар: онтологиялық (біз өмір сүріп жатқан әлем жүйелі ме?); онтологиялық-гносеологиялық (біздің білім жүйелі ме және оның жүйелілігі әлемнің жүйелілігіне адекватты ма?); гносеологиялық (таным процесі жүйелі ме және әлемді жүйелі танудың шегі бар ма?); тәжірибелік (адамның түрдендірушілік қызметі жүйелі ме?).

Жүйе термині бір уақытта бір бүтін ретінде де, әрі алға қойылған мақсаттарға жету мүддесімен бір бүтін ретінде жұмыс істейтін өзара байланысқан әр текті элементтердің біріктірілген жиынтығы ретінде де қарастырылатын объект деп түсіндіріледі. Жүйелер құрамы жағынан да, басты мақсаттары жағынан да бір-бірінен айтарлықтай ерекшеленеді. Бұл бүтін нәрсе жеке элементтерде болмайтын қандай да бір қасиетке ие болады.

Жүйеліліктің ең айқын әрі міндетті белгілерін бөліп көрсетуге болады: оның бүтіндігі мен құрылымдалғандығы, оны құрайтын элементтердің өзара байланыстылығы және бүкіл жүйенің құрылуының белгілі бір мақсатқа бағыныштылығы.

Жүйелілік белгілері үш принциппен сипатталады:

- Сыртқы бүтіндік – дербестік немесе жүйенің қоршаған ортадағы салыстырмалы дербестігі;
- Ішкі бүтіндік – жүйенің қасиеттері оның элементтерінің қасиеттері мен олардың арасындағы өзара байланыстарға тәуелді. Осы байланыстардың бұзылуы салдарынан жүйе өз қызметін атқара алмай қалуы мүмкін;
- Иерархиялылық – жүйеде түрлі ішкі жүйелер болады, басқа жағынан алып қарағанда жүйенің өзі басқа ірі жүйенің ішкі жүйесі болып табылады.

Бүгінгі күні тәжірибеде жүйелік түсініктердің маңыздылығы артып келе жатқаны және адами тәжірибенің жүйелілігі де өсіп келе жатқаны байқалады.

Қоғамдық өндірістің ең маңызды ерекшеліктерінің бірі – оның тиімділігінің үздіксіз өсуі, және, ең алдымен, еңбек өнімділігінің артуы. Еңбек өнімділігінің өсуін қамтамасыз ету –

өте күрделі әрі көп қырлы процесс, бірақ оның нәтижесінде еңбек құралдары мен оны ұйымдастыру әдістері дамиды.

Автоматтандыру еңбек өнімділігін арттырудың қуатты құралы болып табылады.

Мысалы, саланы табысты басқару үшін тірі еңбек шығындары, яғни басқаруға арналған ресурстар қажет, ал адам миының ақпаратты есте сақтау және өңдеу бойынша мүмкіншіліктері шектеулі. Күрделі ақпараттық есептерді тек муниципалды және федералды деңгейдегі әкімшілік органдары шешетін болса, оған мемлекеттің халық санынан 2 есе көп жұмыскерлер қажет болады. Елді басқару кезінде тірі еңбек тапшылығын болдырмау үшін елдің басқару аппаратының әрбір қызметкерінің жұмысының тиімділігін көтеру керек. Бірақ, егер де елді басқару органдарының ақпараттық-аналитикалық жұмысын ЭЕМ-ң көмегімен автоматтандыратын болсақ, мұның қажеті болмай қалады.

Мұнда нені автоматтандыру керектігін түсініп алу маңызды, яғни машинаға тек толығымен зерттелген, егжей-тегжейлі әрі толық сипатталған, әрбір жағдайда нені, қандай ретпен және қалай жасау керектігі анық белгілі, және автоматтың тап болуы мүмкін барлық жағдайлар анық белгілі жұмыстарды ғана толықтай жүктеуге болады. Тек осындай жағдайларда ғана тиісті автоматты құрастыруға болады, және тек осы жағдайларда ғана ол өзіне жүктелген жұмысты сәтті орындай алады.

Күрделі жүйелердегі еңбек өнімділігінің проблемасы автоматтандыру арқылы шешіледі. Осы кезде адамның ақыл-парасаты автоматтандырушы құрылғыларды жасауда пайдаланылады.

Адам әлемді түрлі тәсілдермен игереді. Ең алдымен ол оны сезеді, яғни оны тікелей сезім мүшелері арқылы қабылдайды. Субъектінің жадында сақталатын және оның эмоционалдық күйімен анықталатын мұндай танымның сипаты біз үшін бүтін де, бөлшек те болып табылады – көріністі бүтіндей немесе бөлшектеп көрсетеді. Эмоционалдық күйлер негізінде адамда қоршаған орта туралы түсінік қалыптасады. Бірақ сезімталдық адамда ғана емес, барлық жануарларда да бар. Адамның

ерекшелігі танымның жоғарырақ деңгейі – рационалды таным болып табылады, ол материяның қозғалыс заңдылықтарын тауып, оларды жадыда сақтауға мүмкіндік береді.

Рационалды таным жүйелі болып табылады. Ол реттелген ойлау операцияларынан тұрады және объективтік шынайылық жүйесіне барынша сәйкес келетін ойлау жүйесін қалыптастырады. Адамның тәжірибелік қызметі жүйелі болып табылады, және тәжірибенің жүйелілігінің деңгейі білім артқан сайын және тәжірибе жинақталған сайын арта береді. Материяның және оның қасиеттерінің жалпы жүйелілігі – бұл адамның шынайылықты көрсетуінің және түрлендіруінің әр түрлі түрлерінің жүйелілігі.

Ғылыми таным бей-берекет, ретсіз жүзеге асырылмайды; оның белгілі жүйесі болады және ол белгілі заңдылықтарға бағынады. Осы таным заңдылықтары объективті әлемнің даму және қызмет ету заңдылықтарымен анықталады.

Модельдеу танымның негізгі әдістерінің бірі болып табылады. Модельдеу шынайылықты көрсету формасы болып табылады және шынайы объектілердің, заттардың және құбылыстардың қасиеттерін басқа объектілердің, процестердің, құбылыстардың көмегімен, немесе сурет, жоспар, карта, теңдеулер жиынтығы, алгоритмдер мен бағдарламалар түрінде абстрактілі сипаттаманың көмегімен анықтайды немесе орындайды.

Модельдеу мүмкіншіліктері, яғни модельді құру және зерттеу барысында алынған нәтижелерді түпнұсқаға көшіру мынаған негізделген: модель белгілі бір мағынада объектінің зерттеушіні қызықтыратын кейбір қырларын көрсетеді (орындайды, модельдейді, сипаттайды, имитациялайды).

Бір объектінің (процесті немесе құбылысты) басқа объектіге (процеске немесе құбылысқа) ауыстырылуы, бірақ бастапқы объектінің барлық қасиеттерін сақталуы модельдеу деп аталады, ал объектіні ауыстырушы процестің өзі бастапқы объектінің моделі деп аталады. Модельдің барлық анықтамаларынан мыналарды бөліп көрсетуге болады:

Модель – объект немесе объектінің, жүйенің сипаттамасы, ол түпнұсқаны зерттеу үшін немесе оның қандай да бір

қасиеттерін орындау үшін бір жүйені (яғни түпнұсқаны) басқа жүйеге ауыстыруға арналған.

Модель – бір құрылымды басқа құрылымды көрсету нәтижесі.

Модель – бұл материалды немесе ойша елестетілетін объект, ол оқып зерттеу процесінде түпнұсқа-объектінің орнына жүреді, және берілген зерттеу жұмыстары үшін маңызы бар қырларын сақтайды.

Модельге қойылатын негізгі талаптар: құрылуының көрнекілігі; оның негізгі қасиеттері мен қатынастарының көрінетіндігі; оның зерттеуге және орындауға қолжетімділігі; зерттеудің және орындаудың қарапайымдығы; түпнұсқада болған (гипотезалар моделін құру кезінде дәлдікпен қарастырылатын) ақпараттың сақталуы және жаңа ақпараттың алынуы.

Модельді құру процесі модельдеу деп аталады.

Модельдеу процесуі үш кезеңнен құралады: модельді құру (бұл есепті ресімдеу қиындау және ол конструктивті болып табылады, себебі модельдерді құруға арналған алгоритм жоқ); модельді зерттеу (бұл есепті ресімдеу оңайлау, себебі модельдердің түрлі кластарын зерттеу әдістері бар); модельді қолдану (конструктивті және нақтыланатын есеп).

Модельдеудің мақсаты – өзара және қоршаған ортамен өзара әрекеттесетін объектілер туралы ақпаратты алу, өңдеу, көрсету және қолдану; мұнда модель объектінің қасиеттерін және оның әрекеттерінің заңдылықтарын тану құралы ретінде жүреді.

Модельдеу адам қызметінің түрлі салаларында, әсіресе жобалау және басқару салаларында кеңінен қолданылады, себебі бұл салаларда алынатын ақпарат негізінде тиімді шешімдер қабыладу процестері аса маңызды болып табылады.

Модельдеу теориясы дегеніміз түпнұсқа-объектілерді басқа модель-объектілермен ауыстыру негізінде түпнұсқа-объектілердің қасиеттерін зерттеу тәсілдерін үйретететін ғылым саласы. Модельдеу теориясының негізінде ұқсату теориясы жатыр. Модельдеу кезінде асолютті ұқсастық болмайды және модель объект қызметінің зерттеліп отырған жағын дұрыстап

көрсетсе жеткілікті. Абсолютті ұқсастық тек бір объектіні тура сондай абсқа объектіге ауыстырған жағдайда ғана болады.

Модельдеудің келесідей түрлері ажыратылады: нақты, идеал, физикалық, көрнекі, белгілік, математикалық және т.б.

Берілген кітапта модельдеудің универсал түрі – математикалық модель қарастырылады, ол модельденетін физикалық процеске математикалық қатынастар жүйесін сәйкестендіреді, оны шешу арқылы объектінің әрекеттері туралы сұраққа физикалық модельді құрмастан жауап беруге болады.

Осылайша, математикалық модельдеу – бұл шынайы объектіні, процесті немесе жүйені ЭЕМ көмегімен эксперименталды зерттеуге ыңғайлы математикалық модельге ауыстыра отырып зерттеу құралы болып табылады.

Психология, химия, электротехника, ауыл шаруашылығы, тасымалдарды жоспарлау, басқару, сауда және білім беру сияқты әр түрлі салаларда туындайтын проблемалардың көпшілігін математикалық бағдарламалау есептері ретінде ресімдеуге болады.

«Математикалық бағдарламалау» тіркесі «оңтайландыру» сөзінің синонимы болып табылады. Математикалық бағдарламалаудың көптеген бөлімдері белгілі: сызықтық, бейсызықтық, дискретті, динамикалық, желілік, стохастикалық және т.б. Математикалық бағдарламалаудың әр түрлі бөлімдерінде қарастырылатын есептер математикалық модельдерінің түрімен ерекшеленеді. Есептің оңтайлы шешімі (егер ол бар болса) шешімді табу әдісінің көмегімен табылады. Берілген оқулықта есептерді шешудің көптеген әдістері келтіріледі. Кез келген әдіс алгоритмге дейін нақтыланады. Оңтайлы шешімдердің сандық мәнін анықтау үшін берілетін процедуралар алгоритм деп аталады.

Ақыр аяғында, оңтайландыру берілген жоспарлы шектеулер жағдайында экономикалық ресурстардың шығынын қысқарту және қаржылық нәтижелерді арттыру мақсатында жұмыстар кешенін жоспарлау, ұйымдастыру және басқару процестерін жақсарту болып табылады.

1.3. Операцияларды зерттеудің негізгі ұғымдары мен анықтамалары

«Операцияларды зерттеу» (ОЗ) термині екінші дүниежүзілік соғыс кезінде пайда болды. Сол кезде бұл термин пәннің мазмұнына толықтай сәйкес келген, себебі ОЗ термині әскери операцияларды басқару кезінде математикалық әдістерді қолдану деп түсіндірілетін. ОЗ әдістері ары қарай басқа да салаларда қолданыла бастады, алайда пәннің аты әлі күнге дейін сақталған (Англияда – «операциялық зерттеу», АҚШ-та – «басқару туралы ғылым», ТМД елдерінде – «ОЗ»).

ОЗ-ға ұйымдастырушылық басқару есептерін шешуге арналған ғылыми тәсіл ретінде анықтама беруге болады.

Операция – бір оймен біріктірілген және белгілі бір мақсатқа жетуге бағытталған шаралар жиынтығы.

Бастапқы параметрлерге тәуелді болып келетін кез келген айқындалған таңдау шешім деп аталады. Оңтайлы шешім деп қандай да бір себептермен басқа шешімдерден артық болып табылатын шешімді айтамыз.

ОЗ мақсаты: шешімнің оңтайлылығын алдын ала санмен негіздеу, бірақ шешім қабылдау жауапты тұлғаның құзіреттілігінде болады, оған соңғы таңдау жасау құқығы берілген және жасалған таңдауы үшін жауапкершілік жүктелген.

1.4. Операцияларды зерттеудің негізгі кезеңдері

Операцияларды зерттеу процесі бірнеше кезеңге бөлінеді. Әрбір кезеңде келесі кезеңде қолданылатын жаңа деректер алынып отырады (кейбір жағдайларда алдыңғы кезеңдерге өтуге де болады). Төмендегі кестеде ОЗ кезеңдері көрсетілген; берілген кезеңдегі жұмысты орындайтын мамандар, сонымен бірге ақпараттық жүйелерді жобалау кезеңіне сәйкестігі.

1-кесте

ОЗ кезеңдерінің атауы	**Қажетті мамандар**	**АЖ жобалау кезеңдері**
1	2	3
1. Операцияны (басқару объектісін) жүйелік талдау	Жоспарлаушы (технолог), операция жүргізуші (жүйелік аналитик)	Жоба алдындағы зерттеу
2. Операцияны зерттеу есебін тұжырымдау. Қарапайым актілерді анықтау	Жоспарлаушы (технолог), операция жүргізуші (жүйелік аналитик)	
3. Болжамдарды құру	Жоспарлаушы (технолог), операция жүргізуші (жүйелік аналитик)	Техникалық жобалау
4. Математикалық модельді құру	Операция жүгізуші және қолданбалы математик	
5. Есепті шешу әдісін таңдау және оны шешудің есептік алгоритмін жасау	Операция жүгізуші және қолданбалы математик	
6. Есептік алгоритмді жүзеге асыратын бағдарламаны құру және түзету	Операция жүгізуші, жүйелік бағдарламашы	Жұмыстық жобалау
7. Есепті шешу және есепті шешу нәтижесін талдау	Технолог және операция жүргізуші	Тәжірибелік-өндірістік эксплуатация
8. Есепті шешу нәтижесін практикада жүзеге асыру	Барлық мамандар	Ендіру

1-кезең. Операцияны немесе басқару объектісін жүйелік талдау.

Операцияны бастапқы объектінің үстінен жүргізілетін жеке әрекеттердің жиыны ретінде қарастыруға болады. Бұл әрекеттер (ішкі операциялар) біріге келіп бір процесті (операцияны) құрайды, осының нәтижесінде нәтижеге қол жеткізіледі. Қойылған есепті айқындау процесінде келесілер анықталады:

1) зерттелетін операция үшін қанша ақпарат жинауға болады;

2) соңғы нәтижелер қалай қолданылатын болады;

3) кім шешім қабылдайтын болады және т.б.

Осылайша операция нәтижесіне әсер ететін барлық маңызды факторлар, сонымен бірге осы операция нәтижесін құбылыстардың ауқымдырақ ортасына қаншалықты таратуға болатыны анықталады. Қажетті ақпаратты әдебиеттерді қарау арқылы, құжаттар мен есептемелерді талдау арқылы, эксперименттік мәліметтерді талдау арқылы, мамандар мен сарапшылардан кеңес алу арқылы табуға болады.

Кәсіби түрде жинақталған ақпарат ЭЕМ-ді қолдана отырып алдын ала өңделіп, топталып, сұрыпталуы тиіс. Операцияның мақсаттарын айқындап алған соң сандық критерийлерді анықтау керек, олардың көмегімен есепті шешу нәтижелерін бағалауға болады. Кез келген шешім, егер ол операция нәтижелерінің айтарлықтай жақсаруына септігін тигізсе, жарамды болып табылады.

2-кезең. Операцияны зерттеу есебін тұжырымдау. Қарапайым актілерді анықтау.

Бірінші ішкі кезеңде есептің мазмұнды қойылымы тұжырымдалады. Ол операцияны басқару есебінің математикалық моделін құруға жетерлік болуы керек. Ол зерттелетін операция мен оның ішкі операцияларының физикалық табиғаты мен сандық сипаттамалары туралы, олардың арасындағы өзара әрекеттестік дәрежесі мен сипаты туралы, операцияның жалпы жұмыс жасау процесіндегі әрбір ішкі операцияның орны мен мәні туралы мәліметтерді қамтуы тиіс. Мазмұнды қойылымда зерттеу пәнінің айқын мазмұны, бағаланатын тәуелділіктердің тізімі, келесі кезеңдерде модельді құру кезінде ескерілуі тиіс факторлардың тізімі, сонымен бірге бастапқы деректер болуы тиіс.

Екінші ішкі кезең барлық жағдайларда орындала бермейді, ол тек зерттелетін операцияның күрделілігі себебінен екінші кезеңнен төртінші кезеңге өту мүмкін болмаған кезде ғана орындалады. Бұл ішкі кезеңде берілген операцияның жұмыс жасауына қойылатын негізгі талаптар көрсетіледі. Бұл талаптар әдетте келесі кезеңде бір модельмен сипатталады. Мысалы, өзіндік шектеулер тобын құрайды.

3-кезең. Болжамдарды құру.

Болжамдар зерттелетін операцияны түсіну кезінде кеткен олқылықтардың орнын толтыру үшін қолданылады. Олар күрделі шамаларды қарапайым шамаларға түрлендіру арқылы бастапқы есепті оңайлатуға мүмкіндік береді. Болжамды тұжырымдауға қойылатын талаптар: дұрыс сапалық және сандық қорытынды жасау мүмкіндігі. Болжам тексерілетін болуы тиіс. Болжамның дұрыстығы есепті шешіп оны талдағаннан кейін соңғы рет тексеріледі және ол үлкен дәлдікпен жүргізілуі тиіс.

4-кезең. Математикалық модельді құру.

Айнымалыларына шектеу қойылатын қандай да бір функцияның максимал немесе минимал мәні табылатын есептер математикалық бағдарламалау есептері деп аталады. Максимумы немесе минимумы ізделінетін функция мақсатты функция деп аталады. Шектеулер деп басқарылатын айнымалылардың, яғни мәндерін оңтайландыруға болатын айнымалылардың мүмкін шешу саласын шектейтін арақатынастар жүйесін айтамыз. Басқарылатын айнымалылар арқылы өрнектелген мақсатты функция мен шектеулер есептің математикалық моделін құрайды.

Шектеулерді қанағаттандыратын айнымалы мәндерінің әрбір жиыны мүмкін (тіректі) жоспарды (шешімді) анықтайды, ал мақсатты функцияның максимумын немесе минимумын беретін жиын оңтайлы жоспарды анықтайды. Математикалық модельді құру процесі тәуелсіз және тәуелді айнымалылар тобын құрайтын негізгі айнымалыларды анықтап енгізуден басталады. Тәуелсіз айнымалылар деп есептің шарты бойынша анықталуы тиіс белгісіз жоспарлар немесе басқарушы әсерлерді айтамыз. Тәуелді айнымалылар деп операциялардың күйлерін сипаттайтын және басқарушы әсерлердің көмегімен анықталатын шамаларды айтамыз.

Математикалық модельді құру кезінде келесі принциптер қолданылады:

1. Операциялардың жұмысын және бастапқы деректерді тікелей талдау.

2. Ұқсастықты қолдану тәсілі.

3. Зерттелетін операцияға активті немесе пассивті эксперимент жүргізу.

4. Имитациялық модельдеуді қолдану.

Математикалық модельді құру кезінде келесі талаптар ескеріледі:

1. Адекваттылығы – модель сапасын бағалаудың негізгі критерийі.

2. Әмбебаптығы.

3. Операцияның ескерілетін талаптары мен параметрлерінің санының шектеулілігі.

4. Жақсы болжам жасау қабілеттілігі – модель сапасын бағалау критерийі.

5. Бытыраңқы болмауы (модель өзара айқын байланысы жоқ бөліктерден құралмуы тиіс).

5-кезең. Есепті шешу әдісін таңдау және оны шешудің есептік алгоритмін жасау.

5.1. Әдісті таңдау.

Зерттелетін операцияның сипаты мен параметрлеріне сәйкес ОЗ математикалық модельдерін келесідей белгілері бойынша жіктеуге болады:

1. Ескерілетін ақпарат дәрежесі бойынша:

а) детерминделген оңтайландыру есептері;

б) толық емес ақпарат жағдайындағы оңтайландыру есептері.

2. Айнымалылар саны бойынша:

а) бір өлшемді;

б) көп өлшемді.

3. Шектеулерге (шарттарға) байланысты:

а) шартты экстремум есептері;

б) шартсыз экстремум есептері.

4. Тәуелсіз айнымалылардың уақыттық функция болуына/ болмауына байланысты:

а) статикалық;

б) динамикалық.

Математикалық модельді құрып біткен соң f мақсатты функцияның түріне және X тәуелсіз айнымалылардың сипатына

байланысты оны математикалық бағдарламалау есептерінің кластарының біріне жатқызу керек:

1. Сызықтық бағдарламалау, егер мақсатты функция мен шектеулер сызықтық түрде болса және X айнымалылары мәндер жиынынан соңғы мәндерді қабылдайтын болса;

2. Дискреттік бағдарламалау, егер мақсатты функция мен шектеулер сызықтық түрде болса және X айнымалылары мәндер жиынынан дискретті (бүтін) мәндерді қабылдайтын болса;

3. Бейсызықтық бағдарламалау, егер мақсатты функция немесе шектеулер бейсызықтық түрде болса; дөңес бағдарламалау – егер мақсатты функция дөңес болса және квадраттық бағдарламалау – егер мақсатты функция квадраттық болса;

4. Стохастикалық бағдарламалау, егер айнымалылардың бастапқы мәндері мен мақсатты функция коэффициенттері толық берілмесе;

5. Ойындар және статистикалық шешімдер теориясы, егер активті немесе пассивті қарсылық жағдайында шешім қабылданатын болса;

6. Көп критерийлі оңтайландыру, егер мақсатты функция вектор-функция болса;

7. Динамикалық бағдарламалау, егер тәуелсіз айнымалы белгілі бір t (уақыт, қадам, кезең) параметрінің функциясы болса немесе оны солай көрсетуге болатын болса.

Тұжырымдалған есептің математикалық бағдарламалау есептерінің қай класына жататынына байланысты есепті шешуге қажетті әдістер тобы таңдалады, ал топтың ішінде есептеу алгоритмінің тиімділігіне қойылатын талаптар негізінде нақты әдіс таңдалады.

Тиімділік деп келесідей критерийлер жиынын айтамыз: есепті шешу уақыты (шексіз қадамдары бар әдістер үшін шектілігі немесе жинақтылығы), ЭЕМ жадысының алатын көлемі, шешімнің дәлдік дәрежесі ε. Қолданылатын әдіске және нәтижелердің дәлдігіне қойылатын талаптарға сәйкес келесі әдіс типтерінің бірі таңдалады:

1. Дәл: $\varepsilon = 0$.
2. Жуықталған: $\varepsilon > 0$ – белгілі шама.

3. Эвристикалық: ε – белгісіз шама.

5.2 Есепті шешудің есептік алгоритмін құру.

Оңтайландыру үшін таңдалған кез келген сандық әдіс есептік алгоритмге дейін нақтыланады.

Алгоритм сипаттамасы:

1. Детерминділігі – берілген бастапқы деректермен есептеу процесінде шешімді алудың бірмәнділігі.

2. Есептеу процесінің дискреттілігі – алгоритмнің жеке қадамдарға бөлшектенуі, осы бөлшектердің орындалатындығы күмән тудырмауы тиіс.

3. Көпшілікке ортақтығы – әр түрлі бастапқы деректері бар бір типті есепті шешуге жарамдылығы.

Есептеу алгоритмі блок-схема түрінде көрсетіледі, оның ішінде есептік, логикалық операциялардан және олардың арасындағы өзара байланыстардан тұратын есептеу процесінің реттелген әрі көрнекі көрінісі беріледі.

6-кезең. Есептеу алгоритмін жүзеге асыратын бағдарламаны құру және түзету.

Бағдарламалау процесі келесі қадамдардан тұрады:

1. Есептеу алгоритмінің блок-схемасы бойынша бағдарламаның толық блок-схемасын құру және бағдарлама блок-схемасының дұрыстығын тексеру.

2. Бағдарламалау тілін таңдау.

3. Бағдарламаны құру, бағдарламадағы командалар санын бағалау, ЭЕМ жадысына қойылатын талап, бағдарламаны түзету және бағдарлама мәтінінің бағдарлама блок-схемасына сәйкестігін тексеру.

4. Тестілеу есептерін қолдана отырып бағдарламаны тексеру және қажетті түзетулер енгізу.

5. Есептеу алгоритмінің бір циклына жұмсалатын машина уақытының шығынын бастапқы деректер өлшемінің функциясы ретінде бағалау.

6. Кіріс және шығыс деректерді көрсету формасын дайындау.

Бағдарламаны түзеткен соң бағдарламаға арналған техникалық құжаттама ресімделеді:

1. Қажетті комментарийлері бар бағдарлама мәтіні.

2. Бағдарламаның сипаттамасы.
3. Оны сынау әдістемесі.
4. Қолданушыға арналған нұсқаулық.

7кезең. Есепті шешу және есепті шешу нәтижесін талдау.

7.1 Есепті шешу.

Бұл ішкі кезеңде есепті шешу үшін бастапқы деректер дайындалады, ЭЕМ-де есептеу жүргізіледі, шығыс деректер шешім қабылдаушы тұлғаға (ШҚТ) ыңғайлы формада (кесте, графика және т.б.) алынады.

7.2 Есепті шешу нәтижесін талдау.

ШҚТ көмегімен кіріс және шығыс деректердің сәйкестігі, алынған нәтижелердің толықтығы мен дұрыстығы талданады. Талдап болған соң қорытынды жасалады, олар ұсыныстарда көрсетіледі. Ұсыныстар объективті болуы тиіс, ал олардың тұжырымдалуы анық және дәл болу керек. Егер алынған нәтижелер мен ұсыныстар ШҚТ-ға ұнаса, олар жүзеге асырылады, яғни ендіріледі. Кері жағдайда модельді ары қарай жетілдіру, мәсел есепті қайта тұжырымдау, операцияны қосымша талдау және т.б. туралы шешім қабылданады, яғни ОЗ-ң кез келген алдыңғы кезеңіне қайтады.

8-кезең. Есепті шешу нәтижесін тәжірибеде жүзеге асыру.

ОЗ есептерін шешу нәтижелерін ендіру тек тиісті құзыреттілігі бар басшылар мақұлдаған кезде ғана мүмкін болады.

1.5. Операцияларды зерттеудің негізгі принциптері

1. Жүйелік тәсіл принципі. Белгілі бір операцияны ішкі операциялардың өзара байланыстарын ескере отырып бір бүтін операция ретінде оқып зерттеу және операцияға маңызды факторлардың әсерін зерттеу.

2. Операцияны жүргізушілер құрамының әр түрлі профильділік принципі. Мұндай принципті қолдану адами-машиналық жүйелердің көпшілігінің ОЗ-ң түрлі аспектілерімен байланысты түрлі көзқарасты талап ететіндігіне негізделген. Бұл аспектілерді қандай да бір салада мамандандырылған адамдар түсініп талдай алады.

3. Математикалық модельдеу принципі. Шешім қабылдау үшін математикалық модельдер қолданылады.

4. Оңтайлылық принципі. Шешім қабылдау кезінде біз кез келген шешімге емес, ең жақсы шешімге көңіл бөлуіміз қажет.

5. Іс жүзінде жүзеге асыру принципі. Операцияны басқару есебімен байланысты кез келген зерттеу жұмысы есепті шешу нәтижелерін ендіруге бағытталуы тиіс.

6. Ұқсастық принципі. Жаңа есепті таныс есепке келтіре білу қабілеті.

7. Декомпозиция принципі. Күрделі есепті бірқатар қарапайым есептерге бөлшектеу, сосын бастапқы есеп шешімін алу үшін оларды түйістіру.

8. Біріктіру принципі. Талданатын есепті үлкейту.

1.6. Оңтайландыру әдістерінің жалпы сұлбасы

Әрбір алгоритм жаңа $\overrightarrow{x^{(k+1)}}$ нүктесіне өту мақсатында зерттеу жүргізуге мүмкіндік беретін кезекті $\overrightarrow{x^{(k)}}$ нүктесіне сәйкес анықталатын ережелерді ұсынады. $\overrightarrow{x^{(*)}}$ оңтайландырушы нүктесін іздеу процесі өтулер (қадамдар) тізбегімен сипатталады, бұл ретте олардың әрқайсысы $\overrightarrow{s^{(k)}}$ қозғалыс бағыты туралы және $\lambda^{(k)}$ кезекті қадамының шамасы туралы ақпарат береді.

Алгоритм:

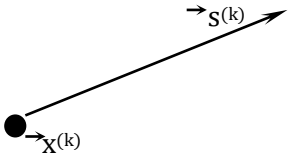
1^0. Бастапқы нүкте (шешім) анықталады $x^{(0)} \in X$, мұндағы

$$X = \{x/y = F(x, a, a_0)\}$$

$$Z = f(x, c) \to \min (\max)$$

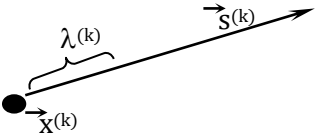
2^0. $\overrightarrow{x^{(k)}} = \overrightarrow{x^{(*)}}$, мұндағы k – итерация нөмірі, $\overrightarrow{x^{(*)}}$ – белгісіз оңтайландырушы шешім. Егер жауап “иә” болса, онда шешім табылды, жауап “жоқ” болса, онда **3^0** қадамға көшеміз.

3^0. $\overrightarrow{s^{(k)}}$ қозғалыс бағыты анықталады. Бағыт мақсатты функция мәнін жақсарту бағытын көрсетеді.



1-сурет

4⁰. $\lambda^{(k)}$ қозғалыс қадамы анықталады.



2-сурет

5⁰. Жаңа шешім анықталады $\overrightarrow{x^{(k+1)}} = \overrightarrow{x^{(k)}} + \lambda^{(k)}\overrightarrow{s^{(k)}}$, $k = k + 1$, 2^0 қадамға көшеміз**.**

Тіректі және оңтайлы шешімдерді анықтауға байланысты оңтайландырудың тіке, қос мағыналы және аралас әдістері ажыратылады.

Тіке әдісте (сұлбасы жоғарыда келтірілген) алдымен тіректі шешім анықталады, сосын ол оңтайлылыққа тексеріледі. Егер ол оңтайлы болмаса, онда келесі тіректі нүктеге өтеді, ондағы мақсатты функция мәні алдыңғы нүктеге қарағанда жақсырақ болады. Бұл процес мақсатты функцияны жақсарту мүмкін болып тұрғанша жалғаса береді. Тіке әдісте максимал мәнге тіректі шешімдер арқылы төменнен жуықтайды, яғни максимал мән – ең жақсы мән.

Қос мағыналы әдісте алдымен оңтайлы нүкте анықталып, оның жарамдылығы тексеріледі. Егер жарамды болмаса, онда оңтайлылық шарттары бұзбай, мүмкін шешу саласына жақын жердегі жаңа нүктеге өтеді. Бұл жағдайда масимал мәнге жоғары жақтан жуықтайды.

Аралас әдісте есепті шешу үшін жоғарыдағы екі әдіс те қолданылады. Бұл жерде мақсатты функцияның оңтайлы мәніне

бір уақытта екі жақтан да жуықтайды. Мақсатты функциялардың сәйкес мәндерінің арасындағы айырмашылық айтарлықтай аз болған кезде есептеу тоқтатылып, тіке әдіспен алынған есеп шешімін ізделінген жуық шешім ретінде алынады.

1.7. Операцияларды зерттеу есептерінің типтік кластары

Осы параграфта қарастырылатын есептер модельдері жалпы есептеу әдістерінің кез келгенінің көмегімен шешіле алады.

Есептерді шешу кезінде алғашқы кезеңге – математикалық модельді құруға және соңғы кезеңге – алынған оңтайлы шешімді жан-жақты талдауға ерекше мән берген жөн.

Математикалық модельді құру айнымалыларды таңдаудан басталады, олардың сандық мәндерінің қосындысын процес нұсқаларының бірі анықтайды. Сондай-ақ математикалық модельдің қарапайымдылығы, демек оны ары қарай талдау қолайлылығы осы айнымалыларды дұрыс таңдауға байланысты. Айнымалыларды таңдаған соң есептің мазмұндық қойылымы бойынша осы айнымалылар қанағаттандыруы тиіс шектеулерді құру керек. Бұл жерде модельге барлық шектеу шарттарының қосылуын және де есеп шартының талап етуінен қатаң болатын бір де бір артық жазылған шектеудің болмауын қадағалау керек.

Соңында мақсатты функция құрылады, ол ең жақсы нұсқаны анықтау критерийін математикалық формада көрсетеді. Математикалық модельді құрып болған соң оны жеңілдету жолдарын қарастырып, есепті шешу үшін қолайлы есептеу әдісін таңдау қажет.

1-есеп. Қоспаның ең жақсы құрамын анықтау (немесе диета туралы есеп)

Жануарларды тамақтандыруға қажетті түрлі азық түрлеріндегі қоректі заттардың құрамы белгілі болсын. Әрбір азық түрінің бағасы белгілі. Әрбір қоректі зат жануарды тамақтандыруға қажетті мөлшерде болатындай етіп тағам мөлшерін анықтау керек, және осы тағам мөлшеріне кететін

жалпы шығындар минимал болу керек. Келесідей белгілеулерді енгізейік:

m – әр түрлі қажетті қоректі заттардың саны;

n – азық түрлерінің саны;

a_{ij} – j-ші азық түрінің бір бірлігінің құрамындағы і-ші қоректі заттың бірлік саны;

b_i – і-ші қоректік затқа деген минималды тәуліктік қажеттілік;

c_j – j-ші азық түрінің бір бірлігінің құны.

Есептің математикалық моделін құру үшін келесідей айнымалыларды таңдап белгілеп аламыз:

x_j – j-ші азық түрінің саны.

Қарапайым актілерді белгілеп алайық:

1. Қажетті килокалорияларды қамтамасыз ету:

$$\sum_{j=1}^{n} a_{ij} x_j \geq b_i, \ i = \overline{1,m}.2.$$

Тағам мөлшеріне жұмсалатын жалпы шығындарды минималдау:

$$z = \sum_{j=1}^{n} c_j x_j \rightarrow \min, \ x_j \geq 0; \ j = \overline{1,n}.$$

2-есеп. Өнімді шығарудың оңтайлы жоспары туралы

Кәсіпорында шығарылатын өнім номенклатурасы n өнім атауынан тұрады. Қолда бар ресурстардың барлық түрлері бойынша технологиялық тұрғыдан жүзеге асырылатындай, әрбір өнім түрін шығаруға қойылатын шектеулерді қанағаттандыратындай және кәсіпорынға ең үлкен табыс әкелетіндей өнім шығару жоспарын құру керек. Келесідей белгілеулерді енгізейік:

a_{ij} – j-ші өнім түрінің бір бірлігін өндіруге кететін і-ші ресурс түрінің шығыны;

b_i – қолда бар ресурстардың толық көлемдері;

c_j – кәсіпорынның j-ші өнім түрін даярлап, сату кезінде алатын табысы;

a_j, A_j – j-ші өнім түрін шығару көлемі бойынша сәйкесінше төменгі және жоғарғы шекаралары.

Есептің математикалық моделін құру үшін келесідей айнымалыларды таңдап белгілеп аламыз:

x_j – j-ші өнімді шығару көлемі.

Қарапайым актілерді белгілеп алайық:

1. Ресурстардың шектеулілігін ескеру:

$$\sum_{j=1}^{n} a_{ij}x_j \le b_i\,,\ i = \overline{1,m}.$$

2. Шығарылатын өнім көлемін шектеу:

$$a_j \le x_j \le A_j,\ \ j = \overline{1,n}.$$

3. Кәсіпорынның табысын максималдау:

$$z = \sum_{j=1}^{n} c_j x_j \to \max.$$

3-есеп. Металл өнімдерінің тасымалын жоспарлау (немесе транспорт есебі)

Өндіріс көлемі a_1, a_2, …, a_m болатын m өндіріс пункті және тұтыну көлемі b_1, b_2, …, b_n болатын n тұтыну пункті бар. c_{ij} – өнім бірлігін i-ші өндіріс пунктінен j-ші тұтыну пунктіне тасымалдау кезінде жұмсалатын шығындар белгілі. Барлық n тұтыну пунктінің қажеттіліктері қанағаттанатындай, және оларды тасымалдауға кететін жалпы шығындар минимал болатындай тасымалдау жоспарын құру керек.

Есептің математикалық моделін құру үшін келесідей айнымалыларды таңдап белгілеп аламыз:

x_{ij} – i-ші өндіріс пунктінен j-ші тұтыну пунктіне тасымалданатын өнім мөлшері.

Қарапайым актілерді белгілеп алайық:

1. Жалпы транспорт шығындарын минималдау:

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

2. Өндірісті шектеу:

$$\sum_{j=1}^{n} x_{ij} \leq a_i , \quad i = \overline{1,m}.$$

3. Қажеттіліктерді қанағаттандыру:

$$\sum_{i=1}^{m} x_{ij} \geq b_j, \quad j = \overline{1,n},$$

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

4-есеп. Белгілеулер туралы есеп

n авиалиния арасында үлестірілуі тиіс n түрлі ұшақ бар. і-ші ұшақты j-ші авиалинияда қолданудан күтілетін c_{ij} тиімділігі белгілі. Барлық ұшақты қолданудан болатын жалпы тиімділік максимал болатындай етіп, ұшақтарды авиалинияларға (әрбір авиалинияға бір ұшақтан) белгілеу керек.

Есептің математикалық моделін құру үшін келесідей айнымалыларды таңдап белгілеп аламыз:

x_{ij} – і-ші ұшақ түрі j-ші авиалинияға белгіленді.

Қарапайым актілерді белгілеп алайық:

1. Жалпы тиімділікті максималдау:

$$z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \rightarrow \max.$$

2. Әрбір ұшақ бір авиалинияға белгіленеді:

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i = \overline{1,n}.$$

3. Әрбір авиалиния бір ұшаққа ғана қызмет көрсетеді:

$$\sum_{i=1}^{n} x_{ij} = 1, \quad j = \overline{1,n}.$$

$$x_{ij} = \begin{cases} 1, \text{ егер і-ұшақ j-авиалинияға белгіленген болса,} \\ 0, \text{ егер і-ұшақ j-авиалинияға белгіленбеген болса.} \end{cases}$$

5-есеп. Материалды оңтайлы етіп пішу туралы есеп (немесе шығындарды минималдау туралы есеп)

Кәсіпорынға m-түрлі партия түрінде қандай да бір жартылай фабрикат келіп түсті, олардың әрқайсысында әрбір партия үшін бірдей мөлшерде сәйкесінше $a1, a2, ..., am$ жартылай фабрикат бірлігі бар. Келіп түскен жартылай фабрикаттан неғұрлым көп мөлшерде бөлшек комплектілерін дайындау керек, комплектілердің әрқайсысына бірінші бөлшек түрінің k_1 бөлшегі, екінші бөлшек түрінің k_2 бөлшегі, ..., m-ші бөлшек түрінің k_m бөлшегі кіруі тиіс.

Жартылай фабрикаттың әрбір бірлігін n-түрлі тәсілмен бөшектеуге болады, және де і-ші партия бірлігін j-ші тәсілмен бөлу кезінде s-ші түрлі a_{ijs} бөлшек пайда болады.

x_{ij} деп і-ші партиядан j-ші тәсілмен бөлшектенетін жартылай фабрикаттар бірлігінің мөлшерін белгілейік, және мұнда і-ші партиядан j-ші тәсілмен бөлшектеу кезінде s-ші түрлі $a_{ijs}x_{ij}$ бөлшек пайда болуы керек. s-ші түрлі бөлшектердің і-ші партиясынан $x_{11} \ldots x_{ij} \ldots x_{mn}$ жоспары бойынша алынатыны:

$$\sum_{j=1}^{n} a_{ijs} x_{ij}$$

ал барлық m партиядан алынатыны:

$$\sum_{i=1}^{m} \sum_{j=1}^{n} a_{ijs} x_{ij}$$

дайын өнімнің әрбір комплектісіне s-ші түрлі k_s бөлшек кіруі тиіс болғандықтан, бөлшек мынандай мөлшердегі комплектілерді құруға мүмкіндік береді:

$$\sum_{i=1}^{m} \sum_{j=1}^{n} a_{ijs} x_{ij} \, / k_s$$

және осылайша, $\|x_{ij}\|$ жоспары бойынша шығаруға болатын толық комплектілер саны төмендегі бөлінділердің ең кішісіне тең:

$$\frac{\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ijs}x_{ij}}{k_1}, \dots, \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ijs}x_{ij}}{k_l} \to \min$$

немесе мынадай шектеулер болғандағы

$$\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ijs}x_{ij} / k_s \geq \xi(s = \overline{1,l}),$$

$$\sum_{j=1}^{n} x_{ij} = a_i, i = \overline{1,m},$$

$$\xi \geq 0, x_{ij} \geq 0, i = \overline{1,m}, j = \overline{1,n}.$$

$$Z = \xi \to \min$$

6-есеп. Егіс алқабын бөлу есебі

m дақылды егу үшін n жер алқабы берілген, олардың аудандары сәйкесінше $a_1, a_2, \dots, a_n$ га, және де і-ші дақылдың j-ші алқаптағы орташа өнімділігі бір гектардан a_{ij} центнерді құрайды, ал і-ші дақылдың 1 центнерінен болатын пайда p_i рубльді құрайды.

Максимал пайда көру үшін әрбір дақылды қай алқапқа себу керектігін анықтау керек, жоспар бойынша і-ші дақылдың кем дегенде b_i центнерін жинау керек.

Есептің математикалық моделін құру үшін келесідей айнымалыларды таңдап белгілеп аламыз:

x_{ij} – і-ші дақыл түрін егуге берілген j-ші алқаптың ауданы.

$$Z = \sum_{i=1}^{m} p_i \sum_{j=1}^{n} a_{ij}x_{ij} \to \max$$
$$\sum_{i=1}^{m} x_{ij} = a_j, j = \overline{1,n},$$
$$\sum_{j=1}^{n} a_{ij}x_{ij} \geq b_i, i = \overline{1,m},$$
$$x_{ij} \geq 0, i = \overline{1,m}, j = \overline{1,n}.$$

1.8. Сызықтық бағдарламалаудың негізгі есебін жазудың әр түрлі формалары

Сызықтық бағдарламалау есебінің (СБЕ) математикалық моделі келесі формалардың бірінде жазылуы мүмкін:

1. Жазудың каноникалық формасы:

$$z = c_1x_1 + c_2x_2 + \cdots + c_nx_n \to \max \qquad (1.1)$$

оған қойылатын шектеулер

$$\begin{gathered} y_1 = a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \le a_1 \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ y_m = a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \le a_m \end{gathered} \qquad (1.2)$$

$$x_j \ge 0,\ j = \overline{1,n}\ \left(x_j >< 0,\ j = \overline{1,n}\right).$$

2. Жазудың стандартты формасы:

$$z = c_1x_1 + c_2x_2 + \cdots + c_nx_n \to \max \qquad (1.3)$$

оған қойылатын шектеулер

$$\begin{gathered} y_1 = -a_{11}x_1 - a_{12}x_2 - \cdots - a_{1n}x_n + a_1 \ge 0 \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ y_m = -a_{m1}x_1 - a_{m2}x_2 - \cdots - a_{mn}x_n + a_m \ge 0 \end{gathered} \qquad (1.4)$$

$$x_j \ge 0,\ j = \overline{1,n}\ \left(x_j >< 0,\ j = \overline{1,n}\right).$$

3. Жазудың матрицалық формасы:

$$CX \to \max \qquad (1.5)$$

оған қойылатын шектеулер

$$AX \le A_0$$
$$X \ge 0 \qquad (1.6)$$

мұндағы

$$C = (c_1, c_2, \dots, c_n)$$

$$A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \dots & \dots & \dots \\ a_{m1} & \dots & a_{mn} \end{pmatrix}, \quad A_0 = \begin{pmatrix} a_1 \\ \dots \\ a_m \end{pmatrix}, \quad X = \begin{pmatrix} x_1 \\ \dots \\ x_n \end{pmatrix}$$

4. Жазудың қысқартылған формасы:

$$z = \sum_{j=1}^{n} c_j x_j \to \max \qquad (1.7)$$

оған қойылатын шектеулер

$$\sum_{j=1}^{n} a_{ij} x_j \le a_i, \quad i = \overline{1,m},$$
$$x_j \ge 0, \; j = \overline{1,n}. \qquad (1.8)$$

Жазудың каноникалық формасы келесі үш белгісімен сипатталады:

1) теңдеулер жүйесі түріндегі біртекті шектеулер жүйесі;

2) есепте кездесетін барлық айнымалыларға қатысты теріс еместіктің біртекті шарттары;

3) мақсатты функцияны максималдау.

1.6-параграфтағы есептердің математикалық модельдері қысқартылған формада жазылған. Сызықтық есептерді симплекс әдісімен шешу үшін жазудың стандартты формасы қажет. Дербес жағдайларда математикалық модельді жазудың матрицалық формасы ыңғайлы болып табылады.

Сұрақтар мен тапсырмалар

1. “Жүйелік талдау” термині нені білдіреді?
2. Жүйеліліктің негізгі белгілерін атаңыз.
3. Жүйенің негізгі сипаттамаларын атаңыз.
4. Жүйелік түсінікте жүйе тиімділігі дегеніміз не?
5. ОЗ негізгі кезеңдерін атап шығыңыз.
6. ОЗ негізгі принциптерін атап шығыңыз.
7. Тіректі шешім мен оңтайлы шешімді анықтауға байланысты қандай оңтайландыру әдістері бар?
8. ОЗ математикалық модельдері шектеулерге байланысты қалай жіктеледі?
9. ОЗ математикалық модельдерінің ескерілетін ақпарат дәрежесі бойынша жіктелуі?
10. ОЗ математикалық модельдері айнымалылар санына байланысты қалай жіктеледі?
11. ОЗ математикалық моделінің құрамына не кіреді?
12. Математикалық модельге қандай талаптар қойылады?
13. Операцияларды зерттеудің негізгі міндеттерін атаңыз.
14. Математикалық модельді құру кезінде қолданылатын негізгі принциптерді атаңыз.
15. Алгоритмнің негізгі сипаттамаларын атап шығыңыз.
16. Оңтайландыру әдістерінің жалпы сұлбасын келтіріңіз.
17. Сызықтық бағдарламалаудың негізгі есебін жазудың қандай формалары бар?
18. Диета туралы есептің математикалық моделінің тәжірибелік жүзеге асырылуына мысал келтіріңіз.
19. Оңтайлы ассортиментті анықтау есебінің математикалық моделінің тәжірибелік жүзеге асырылуына мысал келтіріңіз.
20. Транспорт есебінің математикалық моделінің практикалық жүзеге асырылуына мысал келтіріңіз.
21. Өндірістің жоспарлануы. 1:2:3 қатынасымен кәсіпорын өнімінің үш түрін (А, Б, В) шығаруы керек, саны 200 бірлік еңбек ресурстары пайдаланылады және өндірілген өнімінің бір бөлігі өндіріс ішіндегі тұтынуға қолданылады. Кестеде берілген өнімнің бір бірлігін өндіруге жұмсалатын өндіріс ішіндегі еңбек шығынының коэффициенттері туралы мәліметтер берілген.

Ресурстар	Шығын шамасы		
	А	Б	В
А	-	0.5	0.2
Б	-	-	0.2
Еңбек	2	3	1

Соңғы өнімді максималдау шарты бойынша кәсіпорынның тиімді жоспарын анықтау керек.

22. Кәсіпорын екі түрлі А және В автомобиль бөлшегін шығарады. Кәсіпорын бұрғыланатын, жонылатын, тегістелетін дайын металдарды сатып алады. Кәсіпорынның станок бөлімінің өнімділігін сипаттайтын мәліметтер кестеде берілген.

Станоктар	Әр станоктың өнімділігі (сағ/бөл)		Станок уақытының бағасы (тг)
Жонылатын	25	40	20
Бұрғыланатын	28	35	14
Тегістелетін	35	25	17.5

А бөлшегін шығаратын дайын зат құны 2 теңге, В бөлшегін шығаратын дайын зат құны 3 теңге. А және В бөлшектерін 5 және 6 теңгеден сатады. Өнімді шығарудың тиімді жоспарын табу керек. (бір сағатта әрбір бөлшек түрінің қаншасы шығарылады).

23. А1, А2, А3 теміржол станцияларында 120, 110 және 130 жүк тиелмеген вагондар тұрып қалды. Бұл вагондарды В1, В2, В3, В4 және В5 теміржол бекеттеріне жіберу керек.

$$C = \begin{pmatrix} 2 & 4 & 1 & 6 & 7 \\ 3 & 3 & 5 & 4 & 2 \\ 8 & 9 & 6 & 3 & 4 \end{pmatrix}$$

Әрбір бекеттің вагондарға деген қажеттілігі сәйкесінше 80, 60, 70, 100 және 50 вагонға тең. А2 теміржол бекетінен В2 және В4 бекеттеріне вагондарды жеткізу мүмкін еместігін ескере отырып, және бір вагонның жеткізілу тарифі С матрицасымен анықталатынын біле тұра, тасымалдаудың жалпы құны минимал болатын жоспарды құру керек.

24. Үш жер телімі бар, оларға жүгері, бидай, арпа, тары егілуі мүмкін. Әр жер телімінің ауданы 600, 180 және 220 га тең. Жүгері, бидай, арпа, тары тұқымдарының бар болуын ескеріп, бұл дақылдар 290, 180, 170 және 420 га жерге себілуі қажет. Әр жер теліміндегі дақылдың түсімі әр түрлі және ол С матрицасымен беріледі:

$$C = \begin{pmatrix} 40 & 45 & 50 \\ 30 & 28 & 22 \\ 18 & 22 & 14 \\ 24 & 18 & 16 \end{pmatrix}$$

Жалпы астық түсімі максимал болатындай етіп, әрбір жер теліміне әр дақылдан қанша гектар себу керектігін анықтау керек.

25. Совхозда 6 жыл көлемінде бағыла алатын, 100 басты ірі қара малдан тұратын табын бар. Малдың бір бөлігін сәйкесінше әр жылдың соңында мал басын 600, 580, 560, 540, 520 және 500 теңгеден сатуға болады. Табындағы қалған мал саны келесі жылдың соңында екі есеге көбейеді. Жобаланатын мерзімде (6 жыл) максимал пайда келтіретіндей ең тиімді сату жобасын анықтау керек.

26. Дәрі-дәрмек жасау фирмасы он екі ай ішінде, осы фирманың басты қарсыласы шығарған дәрімен бәсекелес жаңа дәрі жасап шығаруды жоспарлайды. Жаңа дәріні жасап шығару төрт кезеңнен тұрады, және олар ақырын, қалыпты және жылдам қарқынмен өтуі мүмкін. Кезеңдерді өтудің түрлі қарқындары уақыт пен қаражат жұмсалуының түрлі деңгейлеріне сәйкес келеді, олар келесі кестемен берілген (кестедегі бірінші сан – ұзақтық (ай), екінші сан – шығын (мың. тг)).

Кезеңдер / Қарқын	Теориялық зерттеулер	Зертханалық тәжірибелер	Үкіметтің қолдауы	Өтім
Ақырын	5,5	3,6	6,1	5,8
Қалыпты	4,7	2,8	4,1	4,10
Жылдам	2,10	1,12	2,3	3,15

Фирма шығыны 25 мың теңгеден аспайтын шарт бойынша он екі ай ішінде жаңа дәріні жасап шығарудың ең жақсы әдісін табу керек.

27. Мейрамхана басшысы кезекті 4 күнге жұмыс жоспарын жасай отырып, қол сүрткіштің күнделікті қорының қамтамасыз етілуін естен шығармауы керек. Мейрамхананың 4 күн ішіндегі қол сүрткішке деген қажеттілігі мынаған тең: 110, 210, 190, 100 және ол үш түрлі әдіспен қанағаттандырылуы мүмкін:

– бағасы 5 теңге болатын жаңа қол сүрткіштерді пайдалану;

– пайдаланылған қол сүрткіштерді күн соңында жылдам кір жууға өткізу (кір бір түнде жуылады, әрбір қол сүрткіштің жуылу бағасы 1 теңге);

– пайдаланылған қол сүрткіштерді күн соңында кір жуатын орынға өткізу, ондағы кір 24 сағатта жуылады, әрбір қол сүрткіштің жуылу бағасы 0,5 теңге.

Шығынның аз болуын қамтамасыз ететін ең тиімді стратегияны анықтау керек.

28. Мұнай кәсіпорнынан мұнай өңдеу зауытына бір мұнай құбырымен m түрлі мұнай айдалады. Мұнайдың әр түрлі сұрыптары бір-бірімен қосылғанда, c_{ij} шығыны болады. Мұнайды айдағаннан кейін, әр түрлі мұнайдың қосылғандағы жалпы шығынын минималдайтындай әр түрлі мұнай сұрыптарын айдау реттілігін анықтау керек. Төменде $m = 4$ деп алғандағы сандар келтірілген.

i \ j	1	2	3	4
1	x	3	2	5
2	6	x	4	3
3	2	2	x	1
4	4	6	5	x

29. Ұзындығы 3м болатын 100 бөренені 1,5 м, 1 м, 0,75 м өлшемдерімен әрқайсысын үш келтекке кесу керек. Сонымен қатар, әр өлшемді келтектердің саны бірдей болу керек.

Комплекттердің максимальды санын алу үшін, кесудің оптималды жобасын анықтау (әр комплектке үш өлшемді келтектен келеді).

30. Малға берілетін тәуліктік азық мөлшерін жасауда жас пішен (50 кг-нан артық емес) және сүр шөпті (85 кг- нан артық емес) пайдалануға болады. Азық мөлшерінің белгілі нәрлілігі (азық өлшемінің саны 30- дан кем емес) және құрамында құнарлы заттар болуы керек: ақуыз (1 кг-нан артық емес), кальций (100 г-нан артық емес) және фосфор (80 г-нан артық емес).

Келесі кестеде аталған компоненттердің әрбір азықтың (тағамның) 1 кг-дағы мөлшері және осы азықтардың өзіндік құны (тн/кг) туралы мағлұмат көрсетілген:

Компо-ненттер / Өнімдер	Азық мөлшерінің саны	Ақуыз г/кг	Кальций г/кг	Фосфор г/кг	Өзіндік құны тн/кг
Жас пішен	0,5	40	1,25	2	1,2
Сүр шөп	0,5	10	2,5	1	0,8

Өзіндік құны минималды болатындай оптималды азық мөлшерін анықтау керек.

2. СЫЗЫҚТЫҚ БАҒДАРЛАМАЛАУ

2.1. Бірыңғай шығарып тастау әдісі

2.1.1. Сызықтық теңдеулер жүйесін шешу

Сызықтық теңдеулер жүйесі келесі түрде берілсін:

$$y_i = a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n, \quad i = \overline{1,m}, \quad m < n, \quad m = n, \quad m > n, \qquad (2.1)$$

мұндағы y_i – тәуелді айнымалылар, x_j – тәуелсіз (белгісіз) айнымалылар.

Келесі кестені жүйе теңдеулерімен толтырамыз (2.1):

2-кесте

	x_1	x_2	…	x_n
y_1	a_{11}	a_{12}	…	a_{1n}
y_2	a_{21}	a_{22}	…	a_{2n}
…	…	…	…	…
y_m	a_{m1}	a_{m2}	…	a_{mn}

Теңдеулер жүйесін шешу үшін қарапайым жордан шығаруларын қолданамыз.

a_{rs} шешуші элементі бар жордан шығаруларының бір қадамы бастапқы кестені жаңа кестеге келесі бес ереже бойынша ауыстырады:

1°. Шешуші элемент бірлікке ауыстырылады.

2°. Шешуші бағанның қалған элементтері (шешуші элементтен басқалары) өзгеріссіз қалады.

3°. Шешуші жолдың қалған элементтері (шешуші элементтен басқалары) таңбаларын қарама-қарсы таңбаға өзгертеді.

4°. Жаңа a'_{ij} элементтері келесі өрнек бойынша есептеледі (төртбұрыш ережесі):

$$a'_{ij} = a_{ij}a_{rs} - a_{is}a_{rj} \qquad (2.2)$$

5°. Жаңа кестенің барлық элементтері шешуші элементке бөлінеді.

1-мысал

$$y_1 = x_1 - 2x_2 + 3x_3$$
$$y_2 = -x_1 + x_2 + 2x_3$$
$$y_3 = 2x_1 - x_2 - x_3$$

3-кесте

	x_1	x_2	x_3	
y_1	1	-2	3	
y_2	-1	1	**2**	←
y_3	2	-1	-1	

4-кесте

	x_1	x_2	y_2	
y_1	5	-7	3	
x_3	1	-1	1	/ 2
y_3	3	-1	-1	

5-кесте

	x_1	x_2	y_2
y_1	5/2	-7/2	3/2
x_3	1/2	-1/2	1/2
y_3	3/2	-1/2	-1/2

1-мысалда шешуші элементі a_{23}=2 болып табылатын жордан шығаруларының бір қадамы көрсетілген. Шешуші элементті қоршап белгілеп аламыз. Әрбір кезекті итерация алдыңғы кестеден шешуші элементті таңдаудан басталады. Есептеулерді жеңілдету үшін шешуші элемент ретінде 1-ге тең элементті таңдаған ыңғайлы. Егер мұндай таңдау жасау мүмкін болмай қалса, онда дөңгелектеу кезіндегі қателіктерді төмендету үшін шешуші элемент ретінде абсолюттік мәні ең кіші элементті таңдаған дұрыс. 2.1.2–2.1.4 параграфтарда жордан шығаруларын қолданудың түрлі тәсілдерін қарастырамыз.

2.1.2. Кері матрицаны есептеу

(2.1) теңдеулер жүйесін келесі түрде қайта жазайық:

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n = a_i, \quad i = \overline{1,n}, \qquad (2.3)$$

мұндағы a_i – параметрлер.

6-кесте

	x_1	x_2	…	x_n
a_1				
a_2				
…		$A_{(nxn)}$		
a_n				

Бұл әдісте шешуші элементтерді тек бас диагональ бойынша таңдаған ыңғайлы. Егер бастапқы матрицаның анықтауышы 0-ге тең болмаса (яғни $\Delta A \neq 0$), онда n итерациялар жүргізген соң (мұндағы n – матрица реті) кері матрицаны аламыз:

7-кесте

	a_1	a_2	…	a_n
x_1				
x_2				
…		$A^{-1}_{(nxn)}$		
x_n				

2.1.3. Нөлдік жолдарды шығарып тастау

(2.3) теңдеулер жүйесін келесі түрде қайта жазайық:

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n - a_i = 0, \quad i = \overline{1,n} \quad (2.4)$$

8-кестені толтырамыз, 7–кестемен салыстырғанда мұнда бос мүшелер бағаны пайда болды:

8-кесте

	x_1	x_2	…	x_n	1
0	a_{11}	a_{12}	…	a_{1n}	$-a_1$
0	a_{21}	a_{22}	…	a_{2n}	$-a_2$
…	…	…	…	…	…
0	a_{n1}	a_{n2}	…	a_{nn}	$-a_n$

бос мүшелер бағаны

Жордан шығаруларының n қадамын орындаған соң және әрбір қадамнан соң кестенің жоғарғы жағына шығарылған нөлдің астындағы бағанды сызып тастай отырып, (2.4) теңдеулер жүйесінің шешімін аламыз:

9-кесте

	1
x_1	b_1
x_2	b_2
…	…
x_n	b_n

2-мысал

$$2x_1 + 2x_2 - x_3 + x_4 - 4 = 0$$
$$4x_1 + 3x_2 - x_3 + 2x_4 - 6 = 0$$
$$8x_1 + 5x_2 - 3x_3 + 4x_4 - 12 = 0$$
$$3x_1 + 3x_2 - 2x_3 + 2x_4 - 6 = 0$$

10-кесте

	x_1	x_2	x_3	x_4	1
0	2	2	-1	[1]	-4
0	4	3	-1	2	-6
0	8	5	-3	4	-12
0	3	3	-2	2	-6

11-кесте

	x_1	x_2	x_3	0	1
x_4	-2	-2	1	1	4
0	0	-1	[1]	2	2
0	0	-3	1	4	4
0	-1	-1	0	2	2

12-кесте

	x_1	x_2	0	1
x_4	-2	-1	1	2
x_3	0	1	1	-2
0	0	-2	1	2
0	-1	[-1]	0	2

13-кесте

	x_1	0	1	
x_4	1	-1	0	
x_3	1	1	0	/ (-1)
0	-2	-2	2	
x_2	1	1	-2	

14-кесте

	x_1	0	1
x_4	-1	1	0
x_3	-1	-1	0
0	[2]	2	-2
x_2	-1	-1	2

15-кесте

	0	1
x_4	-1	-2
x_3	-1	-2
x_1	1	2
x_2	-1	2

/ 2

16-кесте

	1
x_4	-1
x_3	-1
x_1	1
x_2	1

$x^* = \{1; 1; -1; -1\}$

Тексеру: $2 * 1 + 2 * 1 - (-1) + (-1) - 4 = 0$

$0 = 0$

2.1.4. Гаусс әдісі

Гаусс әдісі бойынша жордан шығаруларының әрбір қадамынан соң кестеден тек шешуші баған ғана емес, шешуші жол да сызып тасталады, бірақ $\mathbf{X_s}$ үшін өрнек алдын ала жазып алынады. Жордан шығаруларының соңғы қадамында $\mathbf{X_i}$ айнымалыларының бірінің мәні анықталады.

17-кесте

	1
x_i	b_i

Қалған айнымалылардың мәндері табылған айнымалылардың мәндерін алдын ала жазып алған өрнектерге қою арқылы табылады.

Жоғарыда келтірілген мысал үшін:

18-кесте

	x_1	x_2	x_3	x_4	1
0	2	2	-1	1	-4
0	4	3	-1	2	-6
0	8	5	-3	4	-12
0	3	3	-2	2	-6

19-кесте

	x_1	x_2	x_3	0	1
x_4	-2	-2	1	1	4
0	0	-1	1	2	2
0	0	-3	1	4	4
0	-1	-1	0	2	2

$$x_4 = -2x_1 - 2x_2 + x_3 + 4$$

20-кесте

	x_1	x_2	0	1
x_3	0	1	1	-2
0	0	-2	1	2
0	-1	-1	0	2

$$x_3 = x_2 - 2$$

21-кесте

	x_1	0	1	
x_2	-2	-2	2	
0	1	1	-2	/ (-1)

22-кесте

	x_1	0	1
0	2	2	-2
x_2	1	1	2

$$x_2 = -x_1 + 2$$

23-кесте

	0	1	
x_1	1	2	/ 2

24-кесте

	1
x_1	1

Осы мысал үшін $x_1=1$. Қалған айнымалылардың мәндері табылған айнымалылардың мәндерін алдын ала жазып алған өрнектерге қою арқылы табылады:

$$x_1 = 1 \Rightarrow x_2 = 1 \Rightarrow x_3 = -1 \Rightarrow x_4 = -1$$

2.2. Сызықтық бағдарламалау есебінің геометриялық мағынасы

Ең қарапайым жағдайда, n = 2 болғанда, сызықтық бағдарламалау есебінің (СБЕ) геометриялық интерпретациясын алу және оны графикалық әдіспен шешу қиын емес. Екі айнымалы болатын жағдай аса үлкен тәжірибелік маңыздылыққа ие емес, алайда оны қарастыру арқылы СБЕ-ң кейбір қасиеттерін, сонымен бірге оны шешудің кейбір әдістерінің геометриялық мағынасын айқындауға болады.

Келесі түрдегі СБЕ берілсін:

$$z = c_1x_1 + c_2x_2 \rightarrow \max \quad (2.5)$$

$$y_1 = a_{11}x_1 + a_{12}x_2 \leq a_1$$

$$y_2 = a_{21}x_1 + a_{22}x_2 \leq a_2 \quad (2.6)$$

$$\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots$$

$$y_m = a_{m1}x_1 + a_{m2}x_2 \leq a_m$$

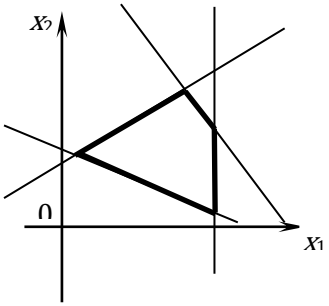
$$x_1 \geq 0,\ x_2 \geq 0. \quad (2.7)$$

(2.5)–(2.7) есептерінің мүмкін шешу саласы не дөңес көпбұрыш (3-сурет), не дөңес көпбұрышты шексіз сала (4-сурет), не болмаса бос сала (5-сурет), немесе жалғыз нүкте болуы мүмкін.

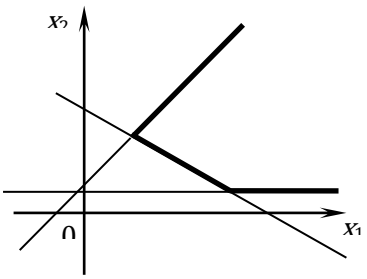
(2.5) теңдеуі x_10x_2 жазықтығында параллель түзулер тобын анықтайды, олардың әрқайсысына z параметрінің белгілі бір мәні жауап береді. Осы түзулерге перпендикуляр $\bar{c} = (c_1, c_2)$ векторы z параметрінің ((2.5) функциясының) барынша тез арту бағытын көрсетеді, ал қарама-қарсы $(-\bar{c})$ векторы барынша тез кему бағытын көрсетеді (6-сурет).

Егер бір координаталар жүйесінде (2.6) – (2.7) есептерінің мүмкін шешу саласын және (2.5) түзулер тобын бейнелейтін болсақ (7-сурет), онда z функциясының максимум нүктесін анықтау есебі мүмкін шешу саласынан z=const тобының түзуі өтетін нүктені табуға келтіріледі, ол z параметрінің ең үлкен мәніне сәйкес келеді.

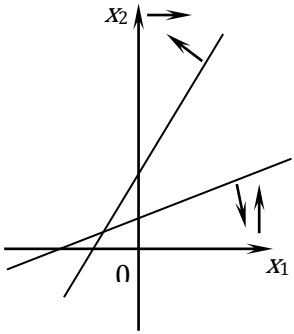
(2.5)–(2.7) есебін практикада шешу үшін мүмкін шешу саласын (МШС), $\bar{c} = (c_1, c_2)$ векторын және оған перпендикуляр z=const түзулер тобының түзуін, мысалы z = 0 тұрғызу керек. Ізделінетін А нүктесі z = 0 көмекші түзуін $\bar{c}$ векторының бағытымен параллель көшіру арқылы табылады. Мүмкін шешу саласының қарама қарсы В төбесінде z функциясы минимум мәнге ие болады (8-сурет).



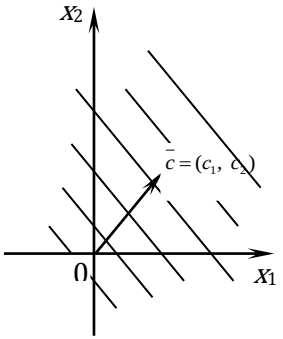
3-сурет



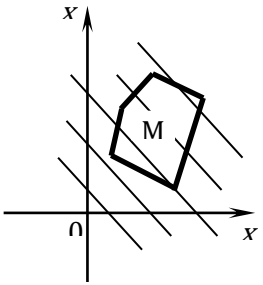
4-сурет



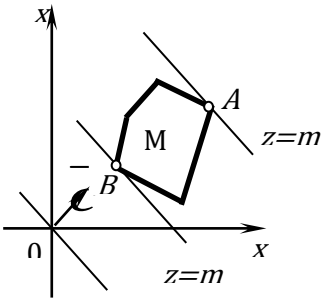
5-сурет



6-сурет



7-сурет



8-сурет

СБЕ-ні графикалық әдіспен шешу алгоритмі:

Декарт жазықтығына тік бұрышты координаталар жүйесін енгізейік те, әрбір (x_1,x_2) сандар жұбын жазықтықтың координаталары x_1 және x_2 болатын нүктесімен салыстырайық.

Алдымен екі айнымалысы бар сызықтық теңсіздіктің біреуін қарастырайық:

$$a_{11}x_1 + a_{12}x_2 \leq a_1$$

Ол $a_1x_1 + a_2x_2 = a$ түзуі бөлетін екі бөліктің (жартылай жазықтықтың) біреуін анықтайды. Сондай-ақ оған сәйкес жартылай жазықтық $a_{11}x_1 + a_{12}x_2 = a_1$ шекаралық түзуін де қамтиды (тұйық жартылай жазықтық). Берілген теңсіздік екі тұйық жартылай жазықтықтың қайсысын анықтайтынын анықтау үшін оған шекаралық түзуде жатпайтын қандай да бір нүктенің координаталарын қойып көру керек. Егер теңсіздік қанағаттандырылса, онда ізделініп отырған жартылай жазықтық сол нүкте жатқан жазықтық болады, ал егер теңсіздік қанағаттандырылмаса, онда оған қарама-қарсы жартылай жазықтық болады.

Сонымен, СБЕ-ң әрбір (2.5) – (2.7) шектеуі жазықтықта қандай да бір жартылай жазықтық анықтайды. Бізге керегі – координаталары барлық теңсіздіктерді (шектеулерді) қанағаттандыратын нүктелер, яғни әрбір жеке шектеу арқылы анықталатын барлық m жартылай жазықтықтарында жататын нүктелер. Демек, СБЕ-ң мүмкін шешімінің жиыны әрбір шектеу арқылы анықталатын жартылай жазықтықтардың қиылысымен (ортақ бөлігімен) геометриялық түрде бейнеленеді. Бұл қиылысты СБЕ-ң мүмкін шешу саласы (МШС) деп атайды.

МШС бос емес болсын делік. Оңтайлы нүктелерді, яғни берілген есептің оңтайлы шешімдеріне сәйкес келетін нүктелерді табу керек. Оңтайлы деп координаталары мақсатты функцияны максимумға жеткізетін МШС-ң ішіндегі нүктелерді айтамыз.

$c_1x_1+c_2x_2$ функциясы $c_1x_1+c_2x_2=a$ түзуінің барлық нүктелерінде бірдей a мәнін қабылдайды, мұндағы a – қандай да бір нақты сан. a-ны параметр деп есептеп, $c_1x_1+c_2x_2$

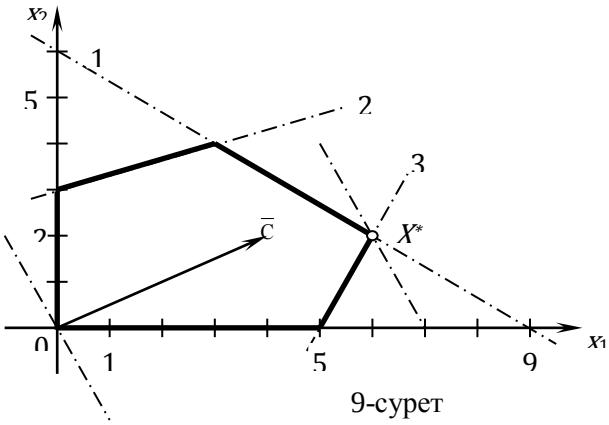
функциясының деңгей сызықтары деп аталатын параллель түзулер тобын аламыз.

Есептің мүмкін шешу саласының max-тағы оңтайлы нүктелерін табу үшін, $c_1x_1+c_2x_2=a$ түзуін МШС-мен қиылысатын кез келген жерінен бастап $c=(c_1,c_2)$ векторының бағытымен ол МШС-мен қиылысып біткенше жылжытып отыру керек. Деңгей сызығын ары қарай жылжытқан кезде бос жиынға кетіп қалатын болсақ, онда сол қалпында ол СБЕ оңтайлы нүктелерінің жиыны болып табылады. Минималдау есебінде деңгей сызығын $c=(c_1,c_2)$ векторының бағытына қарама-қарсы бағытта жылжыту керек.

3-мысал. Сызықтық бағдарламалау есебін графикалық әдіспен шығару:

$$z = 4x_1 + 2x_2 \to \max$$
$$2x_1 + 3x_2 \le 18$$
$$-x_1 + 3x_2 \le 9$$
$$2x_1 - x_2 \le 10$$
$$x_1 \ge 0, \qquad x_2 \ge 0$$

Ш е ш у і .



9-сурет

Теңсіздіктер жүйесінің мүмкін шешу саласын тұрғызайық (9-сурет).

Одан кейін дәл сол координаталар жүйесінде $\overline{C} = (4, 2)$ векторын тұрғызамыз. Координаталар басы арқылы $\overline{C}$ векторына перпендикуляр өтетін түзу z = 0 мәніне сәйкес келетін деңгей сызығы болып табылады. Осы түзуді $\overline{C}$ векторының бағыты бойынша мүмкін шешу саласымен ортақ нүктелері жоқ болғанша өз-өзіне параллель жылжыта отырып, мүмкін шешу саласының ең шеткі жерінде деңгей сызығы X^* нүктесі арқылы өтетіндігін анықтауға болады. Деңгей сызығының осы қалпына $z = z_{max}$ сәйкес келеді. X^* нүктесінің координаталарын табу үшін шекаралас түзулердің теңдеулер жүйесін бірге шешу керек

$$\begin{cases} 2x_1 + 3x_2 = 18 \\ 2x_1 - x_2 = 10 \end{cases}$$

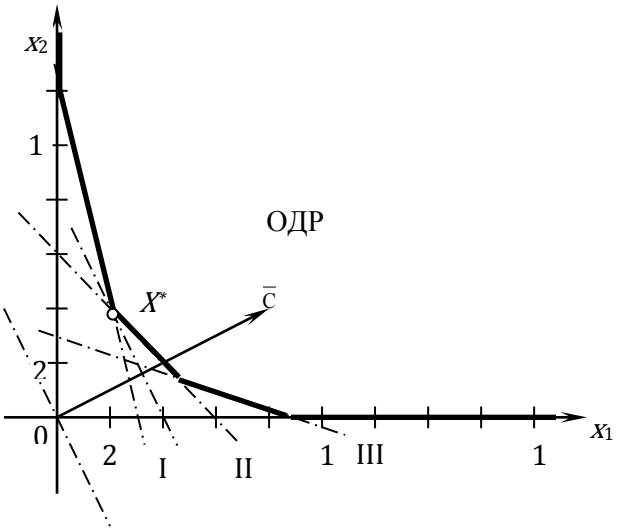
Нәтижесінде алатынымыз – X^* = (6,2). $x_1^* = 6$ және $x_2^* = 2$ мәндерін z функциясына қойып, z_{max} = 28 табамыз.

4–мысал. Сызықтық бағдарламалау есебін графикалық әдіспен шығару:

$$z = 4x_1 + 2x_2 \to \min$$
$$4x_1 + x_2 \geq 12$$
$$x_1 + x_2 \geq 6$$
$$x_1 + 3x_2 \geq 9$$
$$x_1 \geq 0, \qquad x_2 \geq 0$$

Теңсіздіктер жүйесінің мүмкін шешу саласын тұрғызайық (10-сурет).

Ш е ш у і .



10-сурет

Одан кейін дәл сол координаталар жүйесінде $\overline{C} = (4, 2)$ векторын тұрғызамыз. Координаталар басы арқылы $\overline{C}$ векторына перпендикуляр өтетін түзу z = 0 мәніне сәйкес келетін деңгей сызығы болып табылады. Осы түзуді $\overline{C}$ векторының бағытына қарама-қарсы бағытта мүмкін шешу саласымен ортақ нүктелері жоқ болғанша өз-өзіне параллель жылжыта отырып, мүмкін шешу саласының ең шеткі жерінде деңгей сызығы X^* нүктесі арқылы өтетіндігін анықтауға болады. Деңгей сызығының осы қалпына $z = z_{min}$ сәйкес келеді. X^* нүктесінің координаталарын табу үшін шекаралас түзулердің теңдеулер жүйесін бірге шешу керек

$$\begin{cases} 4x_1 + x_2 = 12 \\ x_1 + x_2 = 6 \end{cases}$$

Нәтижесінде X^* = (2, 4) аламыз. $x_1^* = 2$ және $x_2^* = 4$ мәндерін z функциясына қойып z_{min} = 16 аламыз.

Практикалық тұрғыдан СБЕ оңтайлы мәнге ие болатын жағдайлар маңызды болып табылады. n=2 болғанда, егер есептің мүмкін шешу саласы бос емес және шектелген болса (яғни ортасы координаталар басы болып табылатын үлкен радиусты шеңберге толығымен кіретін болса), оның міндетті түрде оңтайлы шешімі болады. Алайда мүмкін шешу саласы шектелмеген болса да, есептің шешімдері болуы мүмкін (4-мысалды қараңыз). Бұл егер максимумға ұмтылатын есеп қарастырылып жатса, есептің мақсатты функциясы мүмкін шешу саласының жоғарғы жағынан шектеулі болғанда, ал минимумға ұмтылатын есеп қарастырылып жатса, төменгі жақтан шектеулі болғанда ғана орындалады. Осылайша, n=2 үшін оңтайлы шешімнің бар болуының қажетті және жеткілікті болып табылатын шарты айқын: Максимумға (минимумға) ұмтылатын СБЕ-ң оның мақсатты функциясы жоғары жақтан (сәйкесінше төменгі жақтан) шектеулі болғанда ғана оңтайлы шешімі бар. Сонымен, келесі тұжырымдар айқын:

1) мүмкін шешу саласы мен оңтайлы нүктелер жиыны – дөңес жиындар;

2) мақсатты функцияның мүмкін шешу саласында шектеулілігі есептің шешілуінің қажетті және жеткілікті шарты болып табылады;

3) есептің оңтайлы нүктесі мүмкін шешу саласының төбесінен табылады.

2.3. Тіке симплекс әдісі

Сызықтық бағдарламалау есебінің математикалық моделі келесі түрде берілген болсын:

$$Z = p_1x_1 + \cdots + p_nx_n \to \max \tag{2.8}$$

$$y_i = -a_{i1}x_1 - a_{i2}x_2 - \cdots - a_{in}x_n + a_i \geq 0,\ i = \overline{1,m} \tag{2.9}$$

$$x_j \geq 0,\ j = \overline{1,n} \tag{2.10}$$

Симплекс әдісінің есептеу сұлбасының негізін түрлендірілген жордан шығарулары құрайды. a_{rs} шешуші элементі бар түрлендірілген жордан шығаруларының (ТЖШ) бір қадамы бастапқы кестені жаңа кестеге келесі бес ережеден тұратын сұлба бойынша ауыстырады:

1) шешуші элемент бірлікке ауыстырылады;

2) шешуші жолдың қалған элементтері (шешуші элементтен басқалары) өзгеріссіз қалады;

3) шешуші бағанның қалған элементтері (шешуші элементтен басқалары) өз таңбаларын қарама-қарсы таңбаға өзгертеді;

4) жаңа a^1_{ij} $(i \neq r, j \neq s)$ элементтері келесі формула бойынша есептеледі:

$a^1_{ij} = a_{ij}a_{rs} - a_{is}a_{rj}$;

5) кестенің барлық элементтері шешуші элементке бөлінеді.

Тіке симплекс әдісінің алгоритмі:

1-ші қадам. Симплекс–кестені толтыру.

25-кесте

	$-x_1$	$-x_2$	…	$-x_n$	1
y_1	$-a_{11}$	$-a_{12}$	…	$-a_{1n}$	$-a_1$
y_2	$-a_{21}$	$-a_{22}$	…	$-a_{2n}$	$-a_2$
...	…	…	…	…	…
y_m	$-a_{m1}$	$-a_{m2}$	…	$-a_{mn}$	$-a_m$
Z	$-p_1$	$-p_2$	…	$-p_n$	0

2-ші қадам. Еркін айнымалыларды шығарып тастау.

Таңбаларына шектеу қойылмаған айнымалылар еркін айнымалылар деп аталады. Ыңғайлы болу үшін барлық x_j $(j = \overline{1,n})$ айнымалыларын еркін айнымалылар деп алайық. ТЖШ-ң n қадамы арқылы барлық xj-лерді кестенің жоғарғы

жолынан оның сол жақ бағанына көшіруге болады, ал олардың орнына y_i-лерді қоюға болады. Келесі кестені аламыз:

26-кесте

	$-y_1$	$-y_2$	…	$-y_n$	1
x_1	b_{11}	b_{12}	…	b_{1n}	b_1
x_2	b_{21}	b_{22}	…	b_{2n}	b_2
…	…	…	…	…	…
x_n	b_{n1}	b_{n2}	…	b_{nn}	b_n
y_{n+1}	$b_{n+1,1}$	$b_{n+1,2}$	…	$b_{n+1,n}$	b_{n+1}
y_m	b_{m1}	b_{m2}	…	b_{mn}	b_m
Z	g_1	g_2	…	g_n	G

x_j $(j = \overline{1,n})$ үшін өрнектерді жазып алып:

$$\begin{cases} x_1 = -(b_{11}y_1 + b_{12}y_2 + \cdots + b_{1n}y_n) + b_1; \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ x_n = -(b_{n1}y_1 + b_{n2}y_2 + \cdots + b_{nn}y_n) + b_n. \end{cases} \quad (2.11)$$

Оларды 19-кестеден сызып тастаймыз. Жұмысты келесі кестемен жалғастырамыз:

27-кесте

	$-y_1$	$-y_2$	…	$-y_n$	1
y_{n+1}	$b_{n+1,1}$	$b_{n+1,2}$	…	$b_{n+1,n}$	b_{n+1}
…	…	…	…	…	…
y_m	b_{m1}	b_{m2}	…	b_{mn}	b_m
Z	g_1	g_2	…	g_n	G

3-ші қадам. Нөлдік жолдарды шығарып тастау.

Шешуші элементтері таңдау ережелері:

1) шешуші баған ретінде b_{k1} оң таңбалы коэффициенті бар баған таңдалады;

2) шешуші жолды таңдау: бос мүшелердің шешуші баған элементтеріне оң мәнді қатынастары есептеледі. Барлық қатынастардың арасынан ең кішісі таңдалады. Осылайша шешуші элемент табылады. ТЖШ-ң бір қадамы орындалады.

4-ші қадам. Тіректі шешімді табу.

Егер 27–кестенің барлық бос мүшелері оң болса, онда сызықтық бағдарламалау есебінің тіректі шешімі табылды деген сөз:

$y_1=y_2=\ldots=y_n=0$,

$y_{n+1}=b_{n+1},\ldots,y_m=b_m$,

$Z^T=G$.

x_j еркін айнымалыларының мәндері (2.11) теңдеулер жүйесінен табылады.

Егер кейбір бос мүшелер теріс таңбалы болса, онда тіректі шешімді табу үшін шешуші элементті таңдаудың келесі ережелері қолданылады:

1) теріс таңбалы бос мүшесі бар жол таңдалады. Егер осы жолдың коэффициенттерінің арасында теріс таңбалы коэффициенттер болмаса, онда (2.9) – (2.10) жүйесі үйлесімсіз;

2) егер қарастырылып отырған жол коэффициенттерінің арасында теріс таңбалылары бар болса, онда олардың кез келген біреуі ($b_{rs}<0$) таңдалып, осы коэффициент тұрған s бағаны шешуші баған болып саналады;

3) шешуші жолды таңдау: бос мүшелердің шешуші баған коэффициенттеріне барлық теріс мәнді емес қатынастары есептеледі ($b_i/b_{is}\geq 0$) де, олардың арасынан ең кішісі $i=i_0$ таңдалады, мұндағы i_0 – шешуші жол. ТЖШ-ң бір қадамы орындалады, және теріс таңбалы бос мүшелер жойылғанша немесе жүйенің үйлесімсіздігі анықталғанша осылай орындала береді.

5-ші қадам. Оңтайлы шешімді табу.

Егер Z-жолдың барлық коэффициенттері оң таңбалы болса, онда есептің оңтайлы шешімі табылды дейміз:

$y_1=y_2=\ldots=y_n=0$,

$y_{n+1}=b_{n+1},\ldots,y_m=b_m$,

$Z^*=G$.

x_j еркін айнымалыларының мәндері (2.11) теңдеулер жүйесінен табылады.

Егер 27-кестенің Z-жолының коэффициенттерінің арасында теріс таңбалылар бар болса, онда оңтайлы шешімді табу үшін шешуші элементті таңдаудың келесі ережелері қолданылады:

1) шешуші баған ретінде Z-жолдың теріс таңбалы коэффициенті бар баған таңдалады (егер теріс таңбалы коэффициенттер бірнеше болса, онда абсолюттік шама бойынша ең үлкен мәнді коэффициенті алынады);

2) шешуші бағанның барлық оң элементтері таңдалып, оларға сәйкес бос мүшелер бөлінеді. Барлық қатынастар арасынан ең кішісі таңдалады. ТЖШ-ң бір қадамы орындалады.

5-мысал.

$Z = -3x_1+6x_2\rightarrow\max$

$y_1 = x_1+2x_2+1\geq 0$

$y_2 = 2x_1+x_2-4\geq 0$

$y_3 = x_1-x_2+1\geq 0$

$y_4 = x_1-4x_2+13\geq 0$

$y_5 = -4x_1+x_2+23\geq 0$

28-кесте

	$-x_1$	$-x_2$	1
y_1	**-1**	2	1
y_2	-2	-1	-4
y_3	-1	1	1
y_4	-1	4	13
y_5	4	-1	23
Z	3	-6	0

29-кесте

	$-y_1$	$-x_2$	1
x_1	-1	2	-1
y_2	-2	**3**	-6
y_3	-1	3	0
y_4	-1	6	12
y_5	4	-9	27
Z	3	-12	-13

$x_1=y_1-2x_2-1$.

Кестеден x_1 айнымалысы бар бірінші жолды сызып тастаймыз.

30-кесте

	$-y_1$	$-y_2$	1
x_2	-2	-1	-6
y_3	3	-3	18
y_4	9	-6	72
y_5	-6	9	27
Z	-15	12	-63

31-кесте

	$-y_1$	$-y_2$	1
x_2	-2/3	1/3	-2
y_3	1	-1	6
y_4	3	-2	24
y_5	-2	3	9
Z	-5	4	-21

$x_2=(2/3)y_1-(1/3)y_2-2$.

Кестеден x_2 айнымалысы бар бірінші жолды сызып тастаймыз.

31-кестеден мәндерді табамыз: $y_1=y_2=0$, $y_3=6$, $y_4=24$, $y_5=9$.

Есептің тіректі шешімі:

$z^T=-21$, $x^T=(3;-2)$.

Оңтайлы шешімді іздеу:

32-кесте

	$-y_1$	$-y_2$	1
y_3	1	-1	6
y_4	3	-2	24
y_5	-2	3	9
Z	-5	4	-21

33-кесте

	$-y_3$	$-y_2$	1
y_1	1	-1	6
y_4	-3	1	6
y_5	2	1	21
Z	5	-1	9

34-кесте

	$-y_3$	$-y_4$	1
y_1	-2	1	12
y_2	3	1	6
y_5	5	-1	15
Z	2	1	15

Соңғы кестеден келесі мәндер табылады:

$y_3=y_4=0$, $y_1=12$, $y_2=6$, $y_5=15$.

x_1 және x_2 айнымалыларының мәндері y_i мәндерін x_j үшін алдын-ала жазып алынған өрнектерге қою арқылы анықталады.

Есептің оңтайлы шешімі:

$z^*=15$, $x^*=(3;4)$.

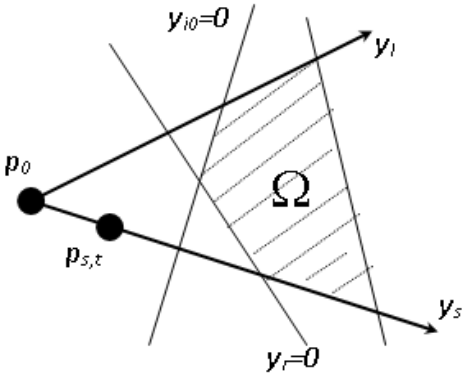
2.4. Шешуші элементті таңдау ережелерін негіздеу

2.4.1. Тіректі шешімді табу кезіндегі шешуші элементті таңдау ережесін негіздеу

Егер кестедегі барлық бос мүшелер нөлден өзгеше болса, онда $p_0\ (y_1 = 0; \ldots; y_n = 0)$ төбесі – координаталардың жаңа төбесі арқылы $y_1=0;\ldots;y_n=0$ тура n жазықтық өтеді [4], және демек, әрқайсысы осы жазықтықтардың (n-1) қиылысында түзілетін $y_1;\ldots;y_n$ (координаталар осьтерінің) тура n қабырғасы бар.

35-кесте

	$-y_1$	…	$-y_s$	…	$-y_n$	1
y_{n+1}	$b_{n+1,1}$	…	$b_{n+1,s}$	…	$b_{n+1,n}$	b_{n+1}
…	…	…	…	…	…	…
y_r	b_{r1}	…	b_{rs}	…	b_{rn}	b_r
…	…	…	…	…	…	…
y_m	b_{m1}	…	b_{ms}	…	b_{mn}	b_m
Z	q_1	…	q_s	…	q_n	Q



11-сурет

$b_r<0$ болсын. Онда $p_0\ (y_1 = 0; \ldots; y_n = 0)$ төбесі Ω көпжағының сыртында жатыр және одан $y_r=0$ жазықтығымен бөлінген (егер Ω 0 болмаса). Шынымен де, $y_r(p_0)=b_r<0$, ал бұл Ω көпжағының нүктелері үшін $y_r\geq 0$ болуы міндетті. Сондықтан p_0 нүктесін Ω-дан бөліп тұрған жазықтықтардың санын азайта отырып, $y_r=0$ жазықтығына қарай жылжуға ұмтылу қалыпты жағдай болып табылады.

Алгоритмнің әрбір қадамы алынған $p_0\ (y_1 = 0; \ldots; y_n = 0)$ төбесінен одан шығатын y_s осьтерінің бірінің бойымен көршілес төбеге дейін жүруді білдіреді. Ось бөліп тұрған жазықтыққа жақындайтындай етіп, яғни нүктенің осы жазықтықтан ауытқуының абсолюттік шамасын азайтатындай етіп таңдалады.

$P_{s,t}$ – y_s осіндегі нүкте, оның координаталары:

$$P_{s,t} = (y_1 = 0; \ldots; y_{s-1} = 0; y_s = t; y_{s+1} = 0; \ldots; y_n = 0)$$

Онда біздің y_s осін таңдауға қоятын талабымыз ($t>0$ – айтарлықтай кішкентай сан) теңсіздік түрінде жазылады:

$|y_r(P_{s,t})| < |y_r(P_0)|$, яғни $|-b_{rs}t + b_r| < |b_r|$, $b_r<0$ болғандықтан, алдыңғы теңсіздік мынаны білдіреді: $-|-b_{rs}t + b_r| < -|b_r|$, $b_{rs}<0$.

$y_s \geq 0$; $t \geq 0$; $b_{rs}<0$

Осыдан анықталғаны: y_s осімен тек s-ші бағанда (нөлден өзгеше $y_s=t$ координатасында) r-жолының теріс коэффициенті болса ғана жылжу керек.

Таңдалған y_s осімен тек $y_{i0}=0$ жазықтығымен алғаш кездескенше ғана жылжыған дұрыс, себебі $y_i=0$ жазықтығымен кездескен кезде: $y_i(P_{s,t}) = 0 = -b_{is}t + b_i$, $b_{is} \neq 0$ болғандықтан, онда $t = \frac{b_i}{b_{is}} \geq 0$ болады, b_i мен b_{is} бірдей таңбалы болуы керек, және сол кезде $i=i_0$ болғанда ең кіші оң мәні алынатын $y_i=0$ жазықтықтарын алғаш кездестіреміз:

$$\frac{b_{i0}}{b_{i0}s} = \min_i \left\{ \frac{b_i}{b_{is}} \geq 0 \right\} = t_0.$$

Мұндай қатынас міндетті түрде табылады, себебі $\frac{b_r}{b_{rs}} > 0$.

Координаталар жүйесіндегі жаңа координатаның барлық төбелері 0-ге тең: $(y_1 = 0; \ldots; y_{s-1} = 0; y_{i0} = 0; y_{s+1} = 0; \ldots; y_n = 0)$.

Ескі жүйеде: $\left(y_1 = 0; \ldots; y_{s-1} = 0; y_s = \frac{b_{i0}}{b_{i0}s}; y_{s+1} = 0; \ldots; y_n = 0\right)$.

Ескерту: Егер $\frac{b_{i0}}{b_{i0}s}$ ең кіші оң қатынасы $b_{i0}<0$, $b_{i0}s<0$ болған кезде алынатын болса (осылай болғаны дұрыс), онда ТЖШ нәтижесінде бір төбеге өтеміз, бұл төбе Ω-дан алдыңғы төбелерге қарағанда азырақ жазықтықтармен бөлінген болады. Басқаша айтқанда, кестедегі теріс таңбалы бос мүшелер саны азаяды, сондай-ақ $|b_r|$ де азаяды.

Ең кіші оң қатынас $b_{i0}>0$ және $b_{i0}s>0$ болған кезде алынса да, кестедегі теріс таңбалы бос мүшелердің саны өзгеріссіз қалса

да, біз $|b_r|$-ді азайтамыз яғни бөліп тұрған $y_r=0$ жазықтығына жақындаймыз.

2.4.2. Оңтайлы шешімді табу үшін шешуші элементті таңдау ережесін негіздеу

z-жол коэффициенттерінің арасында теріс таңбалы мүшелер бар болсын [4], демек, $p_0\ (y_1 = 0; \ldots; y_n = 0)$ төбесінде (11-сурет) z функциясының максимумы алынады деп тұжырымдауға болмайды. Онда p_0 төбесінен бастап y_s осінің бойымен z-тің мәні артатындай етіп жылжу керек, яғни z-тің мәні $P_{s,t} = (y_1 = 0; \ldots; y_{s-1} = 0; y_s = t; y_{s+1} = 0; \ldots; y_n = 0)$ нүктесінде p_0-дегі мәнінен артық болатындай етіп жылжу керек:

$z(p_{s,t})>z(p_0)$

$-q_{s,t}+Q>Q$

$-q_{s,t}>0,\ q_s<0$

Осылайша, z мәнін арттыру үшін кестеде z-жолдың теріс коэффициентінің үстінде тұрған ось бойымен жылжып отыру керек.

$q_s<0$ болсын. p_0 төбесінен бастап көрсетілген ось бойымен $y_i=0$, $i>n$, $b_i\geq 0$ осімен кездескенше жылжып отыру керек (яғни Ω-дан шығып кетпейтіндей етіп, Ω көпжағының көршілес төбесімен кездескенше).

Егер $b_i>0$ болса, онда осы жазықтықпен кездесу нүктесінде алатынымыз:

$$0 = -b_{is}t + b_i;\ \ b_{is} \neq 0;\ \ t = \frac{b_i}{b_{is}} > 0;\ \ b_{is} > 0.$$

Яғни бірінші болып $y_{i0}=0$ ($i_0>n$, $b_{i0}>0$) жазықтығын кездестіреміз, бұл жазықтықта $\frac{b_{io}}{b_{io\,s}}$ қатынасы ең кіші оң мәнге ие.

2.4.3. Нөлдік жолдарды шығарып тастау кезінде шешуші элементті таңдау ережесін негіздеу

Әрбір 0-теңдеу n-өлшемді кеңістікте (n-1)-өлшемді жазықтықты – гипержазықтықты анықтайды, және Ω көпжағы n-өлшемді көпжақтың қиылысына айналады, ол шектеуші-теңсіздіктерге сәйкес келетін жартылай кеңістіктердің 0-теңдеулерге сәйкес келетін гипержазықтықтармен қиылысуы нәтижесінде пайда болған, сондықтан Ω осы гипержазықтықтардың әрқайсысында бар болады [4].

Бірнеше қадамнан соң төмендегі кесте алынған болсын:

36-кесте

	$-x_1$	…	$-x_l$	$-y_1$	…	$-y_h$	1
x_{l+1}	$b_{l+1,1}$	…	…	…	…	$b_{l+1,l+h}$	b_{l+1}
x_s	$b_{s,1}$	…	$b_{s,l}$	$b_{s,l+1}$	…	$b_{s,l+h}$	…
y_{h+1}	…	…	…	…	…	…	…
y_r	…	…	…	…	…	…	…
0	$b_{k,1}$	…	$b_{k,l}$	$b_{k,l+1}$	…	$b_{k,l+h}$	b_k
0	$b_{m,1}$	…	$b_{m,l}$	$b_{m,l+1}$	…	$b_{m,l+h}$	b_m
Z	q_1	…	q_l	q_{l+1}	…	q_{l+h}	Q

0-теңдеулердің біреуін қарастырайық:

$$0 = -(b_{k1}x_1 + \cdots + b_{kl}x_l + b_{kl+1}y_1 + \cdots + b_{kl+h}y_h) + b_k$$

,

мұндағы $b_k > 0$ (*)

Онда $p_0(x_1 = 0; \ldots; x_h = 0; y_1 = 0; \ldots; y_h = 0)$ нүктесі (*) жазықтығынан тыс орналасқан, себебі $b_k \neq 0$. Шешімді алу үшін (ол Ω төбесі және 0-теңдеуін қанағаттандыратын болуы керек) p_0-ден бастап (*) жазықтығына қарай жылжимыз, мысалы, x_l осінің бойымен. Онда

$p_{l,t}(x_1 = 0; \ldots; x_{l-1} = 0; x_l = t; y_1 = 0; \ldots; y_h = 0)$ нүктесі (*) жазықтығына p_0-ге қарағанда жақынырақ болуы керек, яғни $-b_{kl}t + b_k < b_k$ теңсіздігі орындалуы тиіс ($p_{l,t}$ нүктесінің жазықтықтан ауытқуы p_0 нүктесінің ауытқуынан аз болуы керек). Осыдан шығатыны: $-b_{kl}t < 0,\ t > 0,\ b_{kl} > 0$.

Яғни кестеде 0-жолдың оң таңбалы коэффициентінің үстінде орналасқан ось бойымен жылжыған дұрыс. Алдыңғы жағдайлардағыдай, жаңа жазықтықты алғаш кездестіргенше жылжып отыру керек.

Сұрақтар мен тапсырмалар

1. Қарапайым жордан шығаруларының қадамы қандай ережелерден тұрады?
2. Қарапайым жордан шығарушылар мен түрлендірілген жордан шығаруларының айырмашылығы неде?
3. Тіке симплекс әдісінде нөлдік жолдарды шығарып тастау кезінде шешуші элементті таңдаудың қандай ережелері қолданылады?
4. Тіке симплекс әдісімен тіректі шешімді табу кезінде шешуші элементті таңдау ережелерін атаңыз?
5. Тіке симплекс әдісімен оңтайлы шешімді табу кезінде шешуші элементті таңдау ережелерін атаңыз?
6. Қандай жағдайда сызықтық бағдарламалау есебінің геометриялық интерпретациясын келтіруге болады?
7. Қандай айнымалылар еркін айнымалылар болып табылады?
8. Есепті Гаусс әдісімен шығарыңыз:

$$\left.\begin{aligned} 2x_1 + 3x_2 + 11x_3 + 5x_4 &= 5 \\ x_1 + x_2 + 5x_3 + 2x_4 &= 3 \\ 3x_1 + 2x_2 + 8x_3 + 4x_4 &= 5 \\ 3x_1 + 4x_2 + 14x_3 + 9x_4 &= 4 \end{aligned}\right\}.$$

9. Есепті нөлдік жолдарды шығарып тастау әдісімен шешіңіз:

$$\left.\begin{aligned} x_1 + 2x_2 + x_3 &= 8 \\ -2x_1 + 3x_2 - 3x_3 &= -5 \\ 3x_1 - 4x_2 + 5x_3 &= 10 \end{aligned}\right\}.$$

10. Есепті тізбектей шығарып тастау әдістерінің бірімен шешіңіз:

$$\left.\begin{aligned} x_1 + 2x_2 + 3x_3 - 2x_4 &= 6 \\ 2x_1 + 4x_2 - 2x_3 - 3x_4 &= 18 \\ 3x_1 + 2x_2 - x_3 + 2x_4 &= 4 \\ 2x_1 - 3x_2 + 2x_3 + x_4 &= -8 \end{aligned}\right\}.$$

11. Есепті графикалық әдіспен шығарыңыз:

$$z = 4x_1 + x_2 \to \max ,$$

$$\left.\begin{aligned} 4x_1 + 5x_2 &\le 44 \\ 4x_1 - 3x_2 &\le 12 \\ -5x_1 + 4x_2 &\le 20 \end{aligned}\right\},$$

$$x_1 \ge 0,\ x_2 \ge 0.$$

12. Есепті графикалық әдіспен шығарыңыз:

$$z = x_1 + 4x_2 \to \max$$

$$\left.\begin{aligned} -3x_1 + 4x_2 &\le 12 \\ 5x_1 + 4x_2 &\le 44 \\ 4x_1 - 5x_2 &\le 20 \end{aligned}\right\},$$

$$x_1 \ge 0,\ x_2 \ge 0;$$

13. Есепті графикалық әдіспен шығарыңыз:

$$z = 2x_1 + 5x_2 \to \max$$
$$\left.\begin{array}{l} x_1 - x_2 \ge -2 \\ 2x_1 + 3x_2 \le 21 \\ 3x_1 - 5x_2 \le 15 \end{array}\right\},$$
$$x_1 \ge 0,\ x_2 \ge 0;$$

14. Есепті тіке симплекс әдісімен шығарыңыз:

$$z = 2x_1 - x_2 + 3x_3 - 2x_4 + x_5 \to \max \ .$$
$$\left.\begin{array}{l} -x_1 + x_2 + x_3 = 1 \\ x_1 - x_2 + x_4 = 1 \\ x_1 + x_2 + x_5 = 2 \end{array}\right\},$$
$$x_j \ge 0 \ \ \left(j = \overline{1,5}\right).$$

15. Есепті тіке симплекс әдісімен шығарыңыз:

$$z = 2x_1 + x_2 - 3x_3 + 5x_4 \to \max \ ,$$
$$\left.\begin{array}{l} x_1 + 7x_2 + 3x_3 + 7x_4 \le 46 \\ 3x_1 - x_2 + x_3 + 2x_4 \le 8 \\ 2x_1 + 3x_2 - x_3 + x_4 \le 10 \end{array}\right\},$$
$$x_j \ge 0 \ \ \left(j = \overline{1,4}\right).$$

16. Есепті тіке симплекс әдісімен шығарыңыз:

$$z = x_1 + 2x_2 - x_3 \to \max$$
$$\left.\begin{array}{l} -x_1 + 4x_2 - 2x_3 \le 6 \\ x_1 + x_2 + 2x_3 \ge 6 \\ 2x_1 - x_2 + 2x_3 = 4 \end{array}\right\}$$
$$x_1 \ge 0,\ x_2 \ge 0,\ x_3 \ge 0;$$

17. Есепті тіке симплекс әдісімен шығарыңыз:

$$z = -5x_1 + x_2 - x_3 \to \max$$
$$\left.\begin{array}{l} 3x_1 - x_2 - x_3 = 4 \\ x_1 - x_2 + x_3 - x_4 = 1 \\ 2x_1 + x_2 + 2x_3 + x_5 = 7 \end{array}\right\},$$
$$x_j \geq 0 \quad \left(j = \overline{1,5}\right)$$

18. Есепті тіке симплекс әдісімен шығарыңыз:

$$z = 3x_1 - 2x_2 - 5x_4 + x_5 \to \max$$
$$\left.\begin{array}{l} 2x_1 + x_3 - x_4 + x_5 \leq 2 \\ x_1 - x_3 + 2x_4 + x_5 \leq 3 \\ 2x_2 + x_3 - x_4 + 2x_5 \leq 6 \\ x_1 + x_4 - 5x_5 \geq 8 \end{array}\right\},$$
$$x_j \geq 0 \quad \left(j = \overline{1,5}\right)$$

19. Есепті тіке симплекс әдісімен шығарыңыз:

$$z = -x_1 + 2x_2 - x_3 + x_4 \to \min$$
$$\left.\begin{array}{l} 2x_1 - x_2 - x_3 + x_4 \leq 6 \\ x_1 + 2x_2 + x_3 - x_4 \geq 8 \\ 3x_1 - x_2 + 2x_3 + 2x_4 \leq 10 \\ -x_1 + 3x_2 + 5x_3 - 3x_4 = 15 \end{array}\right\},$$
$$x_j \geq 0 \quad \left(j = \overline{1,5}\right)$$

20. Есепті графикалық әдіспен шығарыңыз:

$$z = 5x_1 + 3x_2 \to \max$$
$$\left.\begin{array}{l} 3x_1 - 2x_2 \le 9 \\ 4x_1 - 3x_2 \ge -12 \\ 3x_1 + 4x_2 \le 27 \end{array}\right\},$$
$$x_1 \ge 0,\ x_2 \ge 0$$

21. Есепті графикалық әдіспен шығарыңыз:

$$z = 3x_1 + 2x_2 \to \min$$
$$\left.\begin{array}{l} 2x_1 + x_2 \ge 8 \\ x_1 + x_2 \ge 6 \\ x_1 + 3x_2 \ge 9 \end{array}\right\},$$
$$x_1 \ge 0,\ x_2 \ge 0$$

22. Есепті графикалық әдіспен шығарыңыз:

$$z = 2x_1 + 4x_2 \to \min;$$
$$\left.\begin{array}{l} 3x_1 + 2x_2 \ge 11 \\ -2x_1 + x_2 \le 2 \\ x_1 - 3x_2 \le 0 \end{array}\right\},$$
$$x_1 \ge 0, x_2 \ge 0;$$

23. Есепті тіке симплекс әдісімен шығарыңыз:

$$z = x_1 + x_2 \to \max;$$
$$\left.\begin{array}{l} 2x_1 + 4x_2 \le 16 \\ -4x_1 + 2x_2 \le 8 \\ x_1 + 3x_2 \ge 9 \end{array}\right\},$$
$$x_1 \ge 0, x_2 \ge 0;$$

24. Есепті тіке симплекс әдісімен шығарыңыз:

$$z = 2x_1 + x_2 \to \min;$$

$$\left.\begin{array}{l} -x_1 + 2x_2 \le 6 \\ 9x_1 + 4x_2 \le 56 \\ 3x_1 + 5x_2 \ge 4 \end{array}\right\},$$

$$x_1 \ge 0, x_2 \ge 0;$$

25. Есепті тіке симплекс әдісімен шығарыңыз:

$$z = 2x_1 - 3x_2 \to \max;$$

$$\left.\begin{array}{l} -3x_1 + 2x_2 \le 6 \\ x_1 - 4x_2 \le 2 \\ x_1 - x_2 \le 5 \end{array}\right\},$$

$$x_1 \ge 0, x_2 \ge 0;$$

26. Есепті тіке симплекс әдісімен шығарыңыз:

$$z = 4x_1 + 2x_2 \to \max$$

$$\left.\begin{array}{l} 2x_1 + 3x_2 \le 18 \\ -x_1 + 3x_2 \le 10 \\ 2x_1 - x_2 \le 10 \end{array}\right\},$$

$$x_1 \ge 0,\ x_2 \ge 0$$

27. Есепті тіке симплекс әдісімен шығарыңыз:

$$z = 3x_1 + 2x_2 \to \min$$

$$\left.\begin{array}{l} 7x_1 + 2x_2 \ge 22 \\ x_1 + 2x_2 \ge 10 \\ 3x_1 + 12x_2 \ge 36 \end{array}\right\},$$

$$x_1 \ge 0,\ x_2 \ge 0$$

28. Есепті тіке симплекс әдісімен шығарыңыз:

$$z = x_1 + x_2 \to \min$$
$$\left.\begin{array}{l} x_1 + x_2 \geq 1 \\ -x_1 + x_2 \geq 2 \\ -x_1 + x_2 \geq -3 \end{array}\right\},$$
$$x_1 \geq 0,\ x_2 \geq 0$$

29. Есепті тіке симплекс әдісімен шығарыңыз:

$$z = -3x_1 + 12x_2 \to \max$$
$$\left.\begin{array}{l} x_1 + 4x_2 \leq 16 \\ x_1 - x_2 \geq 2 \\ 3x_1 - 5x_2 \leq 8 \end{array}\right\},$$
$$x_1 \geq 0,\ x_2 \geq 0$$

30. Есепті тіке симплекс әдісімен шығарыңыз:

$$z = -3x_1 + 2x_2 \to \max$$
$$\left.\begin{array}{l} -2x_1 + 3x_2 \geq 6 \\ x_1 + 4x_2 \leq 16 \end{array}\right\},$$
$$x_1 \geq 0,\ x_2 \geq 0$$

3. СЫЗЫҚТЫҚ БАҒДАРЛАМАЛАУДАҒЫ ҚОС МАҒЫНАЛЫЛЫҚ

3.1. Қос мағыналы есептер

Тіке есеп		Қос мағыналы есеп	
$Z = p_1x_1 + \cdots + p_nx_n \to \max$	(3.1)	$\omega = a_1u_1 + \cdots + a_mu_m \to \min$	(3.4)
$y_i = a_{i1}x_1 + \cdots + a_{in}x_n \le a_i,\ i = \overline{1,k}$ $y_i = a_{i1}x_1 + \cdots + a_{in}x_n = a_i,\ i = \overline{(k+1),m}$	(3.2)	$u_i \ge 0,\ i = \overline{1,k}$ $u_i \le 0,\ i = \overline{(k+1),m}$	(3.5)
$x_j \ge 0,\ j = \overline{1,l}$ $x_j \le 0,\ j = \overline{(l+1),n}$	(3.3)	$v_j = a_{1j}u_1 + \cdots + a_{mj}u_m \ge p_j, j = \overline{1,l}$ $v_j = a_{1j}u_1 + \cdots + a_{mj}u_m = p_j,\ j = \overline{(l+1),n}$	(3.6)

Қос мағыналы есепті тіке есепке құрастыру ережелері:

1) тіке есептің (3.2) шектеулерінің $a_i (i = \overline{1,m})$ бос мүшелері қос мағыналы есептің (3.4) мақсатты функциясының коэффициенттері болады, ал тіке есептің (3.1) мақсатты функциясының $p_j (j = \overline{1,n})$ коэффициенттері қос мағыналы есептің (3.6) шектеулерінің бос мүшелері болады;

2) қос мағыналы есептің (3.6) шектеулер коэффициенттерінің матрицасы A^T матрицасы болады, ол тіке есептің (3.2) шектеулер коэффициенттерінің A матрицасын транспонирлеу арқылы алынады;

3) тіке есептің әрбір шектеуші-теңсіздігіне қос мағыналы есептің $u_i\ (i = \overline{1,k})$ еркін емес айнымалысы қарама-қарсы таңбамен сәйкес келеді;

4) тіке есептің әрбір шектеуші-теңдігіне қос мағыналы есептің $u_i (i = \overline{(k+1),m})$ еркін айнымалысы сәйкес келеді;

5) тіке есептің әрбір $x_j (j = \overline{1,l})$ еркін емес айнымалысына қос мағыналы есептің шектеуші-теңсіздігі өз таңбасымен сәйкес келеді;

6) тіке есептің әрбір $x_j (j = \overline{(l+1),n})$ еркін айнымалысына қос мағыналы есептің шектеуші-теңдігі сәйкес келеді;

7) тіке есептің (3.1) мақсатты функциясын максималдау қос мағыналы есептің (3.4) мақсатты функциясын минималдауға ауыстырылады.

6-мысал

Тіке есеп	Қос мағыналы есеп
$Z=10x_1-x_2-42x_3-52x_4\rightarrow max$	$\omega=u_1-9u_2+2u_3+7u_4\rightarrow min$
$-3x_1+x_2+4x_3+x_4\leq 1$	$-3u_1+3u_2-2u_3-3u_4\geq 10$
$3x_1-2x_2+2x_3-2x_4\leq -9$	$u_1-2u_2+u_3+2u_4\geq -1$
$-2x_1+x_2+x_3+3x_4=2$	$4u_1+2u_2+u_3-3u_4\geq -42$
$-3x_1+2x_2-3x_3=7$	$u_1-2u_2+3u_3\geq -52$
$x_i\geq 0$, i=1,4	$u_1\geq 0$, $u_2\geq 0$, $u_3>0$, $u_4>0$

3.2. Қос мағыналы кестелер

Келесі теңдеулер жүйесін қарастырайық

$$y_i = \sum_{j=1}^{n} a_{i,j}(-x_j),\ i = \overline{1,m} \qquad (3.7)$$

және оған кесте құрайық:

37-кесте

	$-x_1$	$-x_2$	…	$-x_n$
y_1	a_{11}	a_{12}	…	a_{1n}
y_2	a_{21}	a_{22}	…	a_{2n}
...	…	…	…	…
y_m	a_{m1}	a_{m2}	…	a_{mn}

Түрлендірілген жордан шығаруларының бір қадамын орындап, y_r тәуелді айнымалысы мен x_s тәуелсіз

айнымалысының орындарын ауыстырамыз. Келесі кестені аламыз:

38-кесте

	$-x_1$	…	$-y_r$	…	$-y_n$
y_1	b_{11}	…	$-a_{1r}$	…	b_{1n}
…	…	…	…	…	…
x_s	a_{s1}	…	1	…	a_{sn}
…	…	…	…	…	…
y_m	b_{m1}	…	$-a_{mr}$	…	b_{mn}

Енді келесі сызықтық формаларды:

$$v_j = \sum_{i=1}^{m} a_{ij}^{*} u_i, \quad j = \overline{1,n} \tag{3.8}$$

А матрицасына транспонирленген А* матрицасымен қарастырайық. Бұл жүйенің кестесін u_i тәуелсіз айнымалылары сол жақта, ал v_j тәуелді айнымалылары жоғарғы жақта тұратындай етіп құрастырайық. Келесі кестені аламыз

39-кесте

	v_1	…	v_s	…	v_n
u_1	a_{11}	…	a_{1s}	…	a_{1n}
…	…	…	…	…	…
u_r	a_{r1}	…	a_{rs}	…	a_{rn}
…	…	…	…	…	…
u_m	a_{m1}	…	a_{ms}	…	a_{mn}

39-кестенің матрицасы 37-кестенің матрицасымен сәйкес келеді. 39- және 37-кестелер қос мағыналы кестелер деп аталады. u_r және v_s айнымалыларының рөлдерін ауыстыра

отырып, a_{rs} шешуші элементі бар қарапайым жордан шығаруларының бір қадамының көмегімен келесі кестені аламыз:

40-кесте

	v_1	…	u_r	…	v_n	
u_1	b_{11}	…	$-a_{1r}$	…	b_{1n}	
…	…	…	…	…	…	
v_s	a_{s1}	…	1	…	a_{sn}	/ a_{rs}
...	…	…	…	…	…	
u_m	b_{m1}	…	$-a_{mr}$	…	b_{mn}	

Осылайша, негізгі 39-кестеге қолданылған түрлендірілген жордан шығаруларының қадамы қос мағыналы 37-кестеге қолданылған қарапайым жордан шығаруларының қадамына пара-пар.

Осымен өзара қос мағыналы 38- және 39-кестелерді бір кестеге келесідей біріктірудің пайдалылығы анықталды:

41-кесте

		v_1	…	v_s	…	v_n
		$-x_1$	…	$-x_s$	…	$-x_n$
u_1	y_1					
…	…					
u_r	y_r			$A_{(mxn)}$		
...	…					
u_m	y_m					

себебі жордан шығаруларының әрбір қадамы бір уақытта екі кестені де қажетті түрде түрлендіреді.

Қос мағыналы кестелерді қолдан отырып, 3.1-параграфтағы 6-мысалды шешейік:

42-кесте

		v_1	v_2	v_3	v_4	w	
		$-x_1$	$-x_2$	$-x_3$	$-x_4$	1	
u_1	y_1	-3	**1**	4	1	1	←
u_2	y_2	3	-2	2	-2	-9	
u_3	0	-2	1	1	3	2	
u_4	0	-3	2	-3	0	7	
1	z	-10	1	42	52	0	
					↑		

1) алғашқы нөлдік жолды шығарып тастау:

43-кесте

		v_1	u_1	v_3	v_4	w
		$-x_1$	$-y_1$	$-x_3$	$-x_4$	1
v_2	x_2	-3	1	4	1	1
u_2	y_2	-3	2	10	0	-7
u_3	0	1	-1	-3	2	1
u_4	0	3	-2	-11	-2	5
1	z	-7	-1	38	51	-1

2) a_{31}-ді шешуші элемент ретінде таңдап, келесі кестені аламыз:

44-кесте

		u_3	u_1	v_3	v_4	w
		0	$-y_1$	$-x_3$	$-x_4$	1
v_2	x_2	3	-2	-5	7	4
u_2	y_2	3	-1	1	6	-4
v_1	x_1	1	-1	-3	2	1
u_4	0	-3	[1]	-2	-8	2
1	z	7	-8	17	65	6

u_3-ке арналан өрнекті жазып аламыз:

$$u_3 = 3v_2 + 3u_2 + v_1 - 3v_4 + 7 \qquad (3.9)$$

3) екінші нөлдік жолды шығарып тастау:

45-кесте

	u_3	v_3	v_4	w
	0	$-x_3$	$-x_4$	1
x_2	2	-9	-9	8
y_2	1	[-1]	-2	-2
x_1	1	-5	-6	3
y_1	1	-2	-8	2
z	8	1	1	22

u_4-ке арналған өрнекті жазып аламыз:

$$u_4 = 2v_2 + u_2 + v_1 + u_1 \qquad (3.10)$$

4) Тіректі шешімді табу:

46-кесте

		u_2	v_4	w
		$-y_2$	$-x_4$	1
v_2	x_2	-9	9	26
u_3	x_3	-1	2	2
v_1	x_1	-5	4	13
u_1	y_1	-2	-4	6
1	z	1	-1	20

5) Оңтайлы шешімді табу:

47-кесте

		u_2	v_3	w
		$-y_2$	$-x_3$	1
v_2	x_2			17
u_4	x_4			1
v_1	x_1			9
u_1	y_1			10
1	z	1/2	1/2	21

Жауабы:

$\max z = \max w$

$v_2 = v_4 = v_1 = u_2 = 0 \Rightarrow u_2 = \frac{1}{2}, v_3 = \frac{1}{2} \Rightarrow v^* = \left\{0; \frac{1}{2}; -17; \frac{17}{2}\right\}$

$y_2 = 0, x_3 = 0 \Rightarrow x_2 = 17, x_4 = 1, x_1 = 9, y_1 = 10 \Rightarrow x^* = \{9; 17; 0; 1\}$

3.3. Қос мағыналылық теоремалары

Қос мағыналылықтың бірінші теоремасы.

Егер тіке және қос мағыналы есептердің оңтайлы шешімдері бар болса, онда оңтайлы нүктеде олардың мақсатты функцияларының мәндері сәйкес келеді (max z = max ω). Қалған шешімдер үшін $\mathbf{Z} \leq \mathbf{W}$ шешімі орындалады. Егер қос мағыналы есептер жұбының біреуінің шешімі жоқ болса, онда екіншісінің не шешімі жоқ, не оның мақсатты функциясы шектелмеген.

Қос мағыналылықтың екінші теоремасы.

Егер X^* және U^* қос мағыналы есептер жұбының оңтайлы шешімдері болса, онда олар қатаң еместіктің келесі қосалқы шарттарын қанағаттандырады:

$$x_j^*\left(\sum_{i=1}^{m} a_{ij}u_i^* - p_i\right) = 0,\ j = \overline{1,n}$$
$$u_i^*\left(\sum_{j=1}^{n} a_{ij}x_j - a_i\right) = 0,\ i = \overline{1,m}$$

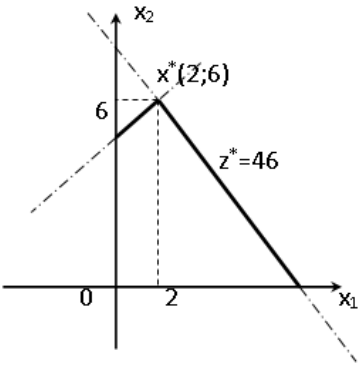
7-мысал

$Z=2x_1+7x_2\rightarrow\max$

$-2x_1+3x_2\leq 14$

$x_1+x_2\leq 8$

$x_1\geq 0,\ x_2\geq 0$



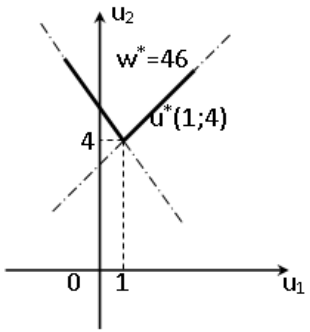
12-сурет

$W=14u_1+8u_2\rightarrow\min$

$-2u_1+u_2\geq 2$

$3u_1+u_2\geq 7$

$u_1\geq 0,\ u_2\geq 0$



13-сурет

$Z^*=w^*=46$; $x^*=(2;6)$; $u^*=(1;4)$.

Қос мағыналылықтың екі теоремасының шарттарын тексеруге болады.

Қос мағыналылық теоремаларының дәлелдері

Тіке және қос мағыналы есептердің шарттарын бір кестеге келесі түрде біріктірейік:

48-кесте

		V_1	…	V_n	W
		$-x_1$	…	$-x_n$	1
U_1	y_1	$-a_{11}$	…	$-a_{1n}$	$-a_1$
…	…	…	…	…	…
U_m	y_m	$-a_{m1}$	…	$-a_{mn}$	$-a_m$
1	Z	$-p_1$	…	$-p_n$	0

Негізгі тіке есепті шеше отырып, қос мағыналы есептің де шешімін табамыз.

Қос мағыналылықтың бірінші теоремасының дәлелі:

49-кесте

		V_1	…	V_l	0	…	0	W
		$-x_1$	…	$-x_l$	$-x_{l+1}$	…	$-x_n$	1
U_1	y_1	$-a_{11}$	…	$-a_{1l}$	$-a_{1,l+1}$	…	$-a_{1n}$	$-a_1$
…	…	…	…	…	…	…	…	…
U_k	y_k	$-a_{k1}$	…	$-a_{kl}$	$-a_{k,l+1}$	…	$-a_{kn}$	$-a_k$
U_{k+1}	0	$-a_{k+1,1}$	…	$-a_{k+1,l}$	$-a_{k+1,l+1}$	…	$-a_{k+1,n}$	$-a_{k+1}$
…	…	…	…	…	…	…	…	…
U_m	y_m	$-a_{m1}$	…	$-a_{ml}$	$-a_{m,l+1}$	…	$-a_{mn}$	$-a_m$
1	Z	$-p_1$	…	$-p_l$	$-p_{l+1}$	…	$-p_n$	0

Негізгі есептің ақырғы шешімі бар болсын, онда (n-l) барлық ($x_{l+1},\dots,x_n$) еркін айнымалыларды шығарып тастаған соң, 1-кесте (n-l) жолға қысқарады, ал барлық 0-жолдардан құтылған соң кесте (m-k) бағанға қысқарады. Алынған қысқартылған матрицадан теріс таңбалы бос мүшелерді және z-жолдағы теріс таңбалы коэффициенттерді жойған соң ақырғы кестені аламыз:

50-кесте

		V_1	…	V_r	U_1	…	U_s	W
		$-x_1$	…	$-x_r$	$-y_1$	…	$-y_s$	1
V_{r+1}	x_{r+1}	$b_{r+1,1}$	…	$b_{r+1,r}$	$b_{r+1,1}'$	…	$b_{r+1,s}'$	b_{r+1}
…	…	…	…	…	…	…	…	…
V_1	x_1	b_{l1}	…	b_{lr}	b_{l1}'	…	b_{ls}'	b_l
U_{s+1}	y_{s+1}	$c_{s+1,1}$	…	$c_{s+1,r}$	$c_{s+1,1}'$	…	$c_{s+1,s}'$	c_{s+1}
…	…	…	…	…	…	…	…	…
U_k	y_k	c_{k1}	…	c_{kr}	c_{k1}'	…	c_{ks}'	c_k
1	Z	q_1	…	q_r	q_1'	…	q_s'	Q

мұндағы $b_{r+1},\dots,b_l,c_{s+1},\dots,c_k\geq 0$,
$q_1,\dots,q_r,q_1`,\dots q_s`\geq 0$
және $x_1^*=\dots=x_r^*=y_1^*=\dots=y_s^*=0$ болғанда
мынаны аламыз: $x_{r+1}^*=b_{r+1}$,
$x_l^*=b_l$,
$y_{s+1}=c_{s+1}$,
$y_k=c_k$,
max Z=Q.

Қос мағыналы есептің кестесін талдайық. $V_{r+1}=\dots=V_l=U_{s+1}=\dots=U_k=0$ деп алып, қос мағыналы есептің тіректі шешімдерін аламыз: $V_1=q_1,\dots,V_r=q_r$, $U_1=q_1`,\dots,U_s=q_s`$.

Қалған U_i бос болып табылады.

$b_{r+1},\ldots,b_l$, $c_{s+1},\ldots,c_k$ теріс емес болғандықтан, мынаны аламыз: $w=b_{r+1}V_{r+1}+\ldots+b_lV_l+c_{s+1}U_{s+1}+\ldots+c_kU_k+Q\square Q$, яғни $V_{r+1}=\ldots=V_l=U_{s+1}=\ldots=U_k=0$ нүктесінде $\min w$ болады, яғни $\min w = Q = \max Z$.

Енді негізгі есептің Z мақсатты шектелмеген болсын, яғни 50-кестеде қандай да бір x_r жоғарғы айнымалысына сәйкес q_r коэффициенті теріс таңбалы болсын және осы бағанның барлық коэффициенттері теріс таңбалы болсын. Онда қос мағыналы есептен шығатыны: $V_r=b_{r+1,r}V_{r+1}+\ldots+b_{lr}V_l+c_{s+1,r}U_{s+1}+\ldots+c_{kr}U_k+q_r\le q_r<0$, қос мағыналы есептің шектеулер жүйесі қарама-қайшы, яғни V_r-дің теріс еместігі $V_{r+1}=\ldots=V_l=U_{s+1}=\ldots=U_k$-ң теріс еместігімен үйлеспейді.

Теореманың салдары.

Қос мағыналы есептер жұбының $x`=(x_1`,x_2`,\ldots,x_n`)$ және $u`=(u_1`,u_2`,\ldots,u_m`)$ мүмкін шешімдері оңтайлы болуы үшін сәйкесінше Z және w мақсатты функцияларының осы шешімдерде бірдей мәнге ие болуы қажетті және жеткілікті:

$$Z(x') = \sum_{j=1}^{n} p_j x_j' = \sum_{i=1}^{m} a_i u_i' = w(u')$$

Ескерту: қос мағыналылықтың негізгі теориясын дәлелдеу процесінде анықталғаны – тіке есепті (max) симплекс әдіспен ТЖШ аппаратының көмегімен шешу кезінде автоматты түрде қос мағыналы есептің (min) шешімі табылады.

Қос мағыналы есепті (min) симплекс әдіспен, бірақ ҚЖШ аппаратының көмегімен шешу кезінде автоматты түрде тіке есептің (max) шешімі табылады.

Қос мағыналылықтың екінші теоремасының дәлелі:

$x^*=(x_1^*,\ldots,x_n^*)$ және $u^*=(u_1^*,\ldots,u_m^*)$ – қарастырып отырған қос мағыналы есеп жұбының оңтайлы шешімі болсын. Мүмкін шешімдер ретінде олар есептер жұбының барлық шектеуші-

теңсіздіктерін және шектеуші-теңдіктерін қанағаттандырады. Онда:

$$\sum_{j=1}^{n} p_j x_j' \le \sum_{i=1}^{n}\left(\sum_{i=1}^{m} a_{ij} u_i'\right)x_j' = \sum_{i=1}^{m}\left(\sum_{j=1}^{n} a_{ij} x_j'\right)u_i' \le \sum_{i=1}^{m} a_i u_i'$$

яғни x_j және u_i – оңтайлы шешімдер, онда 1-ші теорема бойынша

$$\sum_{j=1}^{n} p_j x_j^* = \sum_{i=1}^{m} a_i u_i^*$$

Сондықтан:

$$\left.\begin{aligned}\sum_{j=1}^{n} p_j x_j^* &= \sum_{j=1}^{n}\left(\sum_{j=1}^{n} a_{ij} u_i^*\right)x_j^* \\ \sum_{i=1}^{m} a_i u_i^* &= \sum_{i=1}^{m}\left(\sum_{j=1}^{n} a_{ij} x_j^*\right)u_i^*\end{aligned}\right\}$$

оңтайлы болып табылады.

Бірінші теңдіктен шығатыны: $\sum_{j=1}^{n} x_j^*\left(\sum_{i=1}^{m} a_{ij} u_i^* - p_j\right) = 0$, (бір қосылғыш оң, ал екіншісі теріс және т.б. болған жағдайда, бірақ бізде барлығы оң мәнділер), ал қосынды қосылғыштар нөлге тең болғанда, ал әрбір қосылғыш оң таңбалы көбейтінді болғанда ғана 0-ге тең болады, демек:

$$x_j^*\left(\sum a_{ij} u_i^* - p_j\right) = 0, \ \forall j.$$

Екінші теңсіздік үшін осыған ұқсас дәлел болады.

3.4. Қос мағыналы есептер жұбының экономикалық түсіндірмесі

Осы уақытқа дейін біз өзара қос мағыналы есептер арасындағы тек математикалық байланысты ғана қарастырдық. Мұнда егер бастапқы есептің белгілі бір мазмұнды мағынасы бар болса, онда оған қатысты қос мағыналы есептің де бастапқы есептің мазмұны арқылы анықталатын мағынасы болады.

Айтылған ережеге мысал келтірейік. Өнімнің қандай да бір түрі n түрлі технологиялық тәсілдермен өндірілетін болсын. Уақыт бірлігінде қолданылатын барлық өндірістік факторлардың (шикізаттың, жартылай фабрикаттардың

жабдықтардың, еңбектің және т.б. әр түрлі түрлері) сандарын және осы кезде алынатын дайын өнім санын көрсететін деректер бар. Бұл деректердегі айырмашылық әр түрлі маркалы жабдықты, өзара ауыстырылатын шикізаттың әр түрлі пропорциясын және т.б. қолдану есебінен пайда болуы мүмкін.

Есепте дайын өнімнің жиынтық саны максимал болатындай етіп, түрлі технологиялармен қолданылуы тиіс интенсивтіліктер анықталуы керек.

Қос мағыналы есеп бір өнімі бар өндірістік жоспарлау есебі сияқты түсіндірілуі мүмкін. Әрбір технологияның табысы (шығыны) өлшенетін бірлікпен өлшенетін барлық ингредиенттердің теріс емес салыстырмалы бағаларын табу керек. Бұл бағалар бірлік интенсивтілік кезінде әрбір технологияның жиынтық бағасы тиісті «табыстан» (шығыннан да болуы мүмкін) төмен болмайтындай болуы керек. Мұндай бағалардың барлық мүмкін нұсқаларының ішінен барлық ингредиенттерінің жиынтық бағасы минималды болғаны таңдалады.

Негізгі есеп: Өнім бірлігінің $p_j (j = \overline{1,n})$ бағасы және $a_i (i = \overline{1,m})$ қолда бар ресурстардың мөлшері берілген жағдайда өнімнің бағалық өрнектегі шығарылымын максималдау үшін x_j $(j = \overline{1,n})$ өнімін қанша көлемде және қандай түрде шығару керек?

Қос мағыналы есеп: Ресурстардың $a_i (i = \overline{1,m})$ саны және өнім бірлігінің $p_j (j = \overline{1,n})$ бағасы берілген жағдайда шығындардың жалпы бағасын минималдау үшін әрбір ресурс бірлігінің u_i $(i = \overline{1,m})$ бағасы қандай болуы керек? (u_i айнымалылары бағалар немесе анық емес есептік бағалар деп аталады).

3.5. Қос мағыналы симплекс әдісі

Қос мағыналы симплекс әдісін бағаларды тізбектей анықтау әдісі деп те атайды.

Тіке симплекс әдісінің алгоритмі екі бөлімнен тұрады:

1) теріс таңбалы бос мүшелерден құтылу (тіректі шешімді табу);

2) Z-жолдағы теріс коэффициенттерден құтылу (оңтайлы шешімді табу).

Қос мағыналы симплекс әдісінің алгоритмі кері тәртіппен жүреді:

1) Z-жолдағы теріс коэффициенттерден құтылу (оңтайлы шешімді табу);

2) теріс таңбалы бос мүшелерден құтылу (тіректі шешімді, сонымен қоса оңтайлы шешімді табу).

Қос мағыналы симплекс әдісінің алгоритмі:

1-ші қадам. Симплекс кестені толтыру.

2-ші қадам. Еркін айнымалыларды шығарып тастау.

3-ші қадам. Нөлдік жолдарды шығарып тастау.

Қос мағыналы симплекс әдісінің алгоритмінің алғашқы үш қадамына тіке симплекс әдісінің алгоритмінде сипатталған ережелер қолданылады. (Осы оқулықтың 2.3-параграфын қараңыз).

4-ші қадам. Оңтайлы шешімді табу.

Егер Z-жолдың барлық коэффициенттері оң таңбалы болса, онда нөл Z-ң жоғарғы жағындағы бағасы болып табылады да, бесінші қадамға көшеміз. Кері жағдайда Z-жолдағы теріс коэффициенттерден арылу үшін шешуші элементті таңдаудың келесі ережелері қолданылады:

1) Z-жолдағы теріс таңбалы коэффициенті бар баған қарастылады (мысалы, $g_s<0$);

2) s-ші бағанда қандай да бір $a_{rs}>0$ элементі бар, r жолы шешуші жол ретінде таңдалады. Егер s-ші бағанның барлық элементтері теріс таңбалы болса, онда Z функциясы жоғарғы жақтан шектелмеген, немесе есептің шарттарында қарама-қайшылық бар деген сөз;

3) Z-жол коэффициенттерінің шешуші жолдағы сәйкес элементтерге оң таңбалы емес қатынастары есептеледі. Барлық қатынастар арасынан ең үлкені таңдалады (абсолюттік шамасы бойынша ең кішісі). ТЖШ-ң бір қадамы орындалады.

5-ші қадам. Тіректі шешімді табу (сонымен қоса оңтайлы шешімді табу).

Егер барлық бос мүшелер оң таңбалы болса, онда есептің тіректі (сонымен қоса оңтайлы) шешімі табылды дейміз.

Егер бос мүшелер арасында теріс таңбалылары болса, тіректі (сонымен қоса оңтайлы) шешімді табу кезінде шешуші элементті таңдаудың келесі ережелері қолданылады:

1) шешуші жол ретінде теріс таңбалы бос мүшесі бар жол таңдалады (мысалы, $b_r<0$);

2) осы жолдың барлық теріс таңбалы коэффициенттері таңдалады (егер шешуші жолдың коэффициенттерінің арасында теріс таңбалы коэффициенттер болмаса, онда есептің шарттарында қарама-қайшылық бар деген сөз);

3) Z-жол коэффициенттерінің шешуші жолдан таңдалған сәйкес коэффициенттерге оң таңбалы емес қатынастары есептеледі. Барлық қатынастар арасынан ең үлкені таңдалады (абсолюттік шамасы бойынша ең кішісі). ТЖШ-ң бір қадамы орындалады.

8-мысал.

$Z = 5x_1-x_2-4x_3 \rightarrow max$

$y_1 = x_2+2x_3-9\geq 0$

$y_2 = -x_1+x_2-1\geq 0$

$y_3 = x_1+x_2-3x_3-8\geq 0$

$y_4 = -x_1+x_3+4\geq 0$

$x_i\geq 0$, i=1,3

51-кесте

	$-x_1$	$-x_2$	$-x_3$	1
y_1	0	-1	-2	-9
y_2	1	-1	0	-1
y_3	-1	-1	3	-8
y_4	1	0	-1	4
Z	-5	1	4	0

52-кесте

	$-x_1$	$-y_2$	$-x_3$	1
y_1	-1	-1	-2	-8
x_2	-1	-1	0	1
y_3	-2	-1	3	-7
y_4	1	0	-1	4
Z	-4	1	4	-1

53-кесте

	$-y_4$	$-y_2$	$-x_3$	1
y_1	1	-1	-3	-4
x_2	1	-1	-1	5
y_3	2	-1	1	1
x_1	1	0	-1	4
Z	4	1	0	15

54-кесте

	$-y_4$	$-y_2$	$-y_1$	1
x_3	1	-1	1	-4
x_2	-2	2	1	-19
y_3	-7	4	-1	1
x_1	-2	-1	1	-16
Z	-12	-3	0	-45

55-кесте

	$-y_4$	$-y_2$	$-y_1$	1
x_3	-1/3	1/3	-1/3	4/3
x_2	2/3	-2/3	-1/3	19/3
y_3	7/3	-4/3	1/3	-1/3
x_1	2/3	1/3	-1/3	16/3
Z	4	1	0	15

56-кесте

	$-y_4$	$-y_3$	$-y_1$	1
x_3	-1/3	-1/3	1	-5/3
x_2	2/3	2/3	2/3	-26/3
y_2	7/3	1	1/3	-1/3
x_1	-5/3	-1/3	1/3	21/3
Z	-7	-1	-1/3	-59/3

57-кесте

	$-y_4$	$-y_3$	$-y_1$	1
x_3	1/4	1/4	-1/4	5/4
x_2	-1/2	-1/2	-1/2	26/4
y_2	-7/4	-3/4	-1/4	1/4
x_1	5/4	1/4	-1/4	21/4
Z	21/4	3/4	1/4	59/4

Жауабы:
$X_1^*=21/4$,
$X_2^*=26/4$,
$X_3^*=5/4$,
$Z^*=59/4$.

Сұрақтар мен тапсырмалар

1. Қос мағыналы есепті тіке есепке құрастыру кезінде тіке есептің шектеуші-теңсіздігіне не сәйкес келеді?

2. Қос мағыналы есепті тіке есепке құрастыру кезінде тіке есептің шектеуші-теңдігіне не сәйкес келеді?

3. Қос мағыналы есепті тіке есепке құрастыру кезінде тіке есептің еркін айнымалысына не сәйкес келеді?

4. Қос мағыналы есепті тіке есепке құрастыру кезінде тіке есептің еркін емес айнымалысына не сәйкес келеді?

5. Қос мағыналылықтың негізгі теоремасы және екінші теоремасы.

6. Негізгі және қос мағыналы есептің экономикалық түсіндірмесін келтіріңіз.

7. Қос мағыналы есепті тіке есепке құрастыру:

$$z = 2x_1 + 3x_2 + 5x_3 \to \max$$

$$\left.\begin{array}{r} x_1 + x_2 - x_3 \ge -5 \\ -6x_1 + 7x_2 - 9x_3 \le 4 \\ x_1 + x_2 + 4x_3 = 10 \end{array}\right\},$$

$$x_1 \ge 0,\ x_2 \ge 0,\ x_3 \ge 0.$$

8. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = 4x_1 - 5x_2 - 4x_3 \to \max$$

$$\left.\begin{array}{r} x_2 + 2x_3 \ge 10 \\ -x_1 + x_2 \ge 1 \\ -x_1 - x_2 + 4x_3 \le 1 \\ x_1 - x_3 \le 1 \end{array}\right\},$$

$$x_1 \ge 0,\ x_2 \ge 0,\ x_3 \ge 0.$$

9. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = -x_1 + x_2 - x_3 + x_4 \to \max$$

$$\left.\begin{array}{l} x_1 + 2x_2 - x_3 + 3x_4 = 6 \\ x_2 + 2x_3 - x_4 = 4 \\ 2x_1 + x_3 + x_4 = 6 \end{array}\right\},$$
$$x_j \geq 0 \quad \left(j = \overline{1,4}\right).$$

10. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = 4x_1 + 2x_2 \to \max$$
$$\left.\begin{array}{l} -x_1 + 2x_2 \leq 6 \\ x_1 + x_2 \leq 9 \\ 3x_1 - x_2 \leq 15 \end{array}\right\},$$
$$x_1 \geq 0,\ x_2 \geq 0;$$

11. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = x_1 - 2x_2 \to \min$$
$$\left.\begin{array}{l} x_1 \geq 10 \\ -2x_1 + x_2 \leq 8 \\ x_1 - 2x_2 \geq 12 \\ x_2 \geq 2 \end{array}\right\},$$
$$x_1 \geq 0,\ x_2 \geq 0;$$

12. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = 3x_1 - x_2 - x_3 + x_4 \to \max$$
$$\left.\begin{array}{l} 3x_1 + 5x_2 + x_3 + x_4 = 32 \\ x_1 - 3x_2 - x_3 + x_4 = -8 \end{array}\right\},$$
$$x_j \geq 0 \quad \left(j = \overline{1,4}\right);$$

13. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = -3x_1 + 4x_2 - 6x_3 \to \min$$

$$\left.\begin{array}{l} 2x_1 + 3x_2 - x_3 \geq 8 \\ -3x_1 + 2x_2 - 2x_3 = 10 \\ 5x_1 - 4x_2 + x_3 \geq 7 \end{array}\right\},$$

$$x_j \geq 0 \quad \left(j = \overline{1,3}\right);$$

14. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = 5x_1 - 2x_2 - 6x_3 + 4x_4 + 2x_5 \to \max$$

$$\left.\begin{array}{l} 2x_1 - x_2 - x_3 + 2x_4 \geq 12 \\ 3x_1 + 2x_2 - 2x_3 + 5x_4 + x_6 = 30 \\ -x_1 + 3x_2 + 5x_3 + x_4 \geq 16 \end{array}\right\},$$

$$x_j \geq 0 \quad \left(j = \overline{1,5}\right);$$

15. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = x_1 + 2x_2 - x_3 \to \max$$

$$\left.\begin{array}{l} -x_1 + 4x_2 - 2x_3 \leq 12 \\ x_1 + x_2 + 2x_3 \leq 17 \\ 2x_1 - x_2 + 2x_3 = 4 \end{array}\right\},$$

$$x_j \geq 0 \quad \left(j = \overline{1,3}\right);$$

16. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = x_1 + x_2 \to \max$$

$$\left.\begin{array}{l} -x_1 + 3x_2 \leq 6 \\ 3x_1 - x_2 \leq 6 \end{array}\right\},$$

$$x_1 \geq 0,\ x_2 \geq 0;$$

17. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = 2x_1 + x_2 \to \min$$

$$\left.\begin{array}{l} 3x_1 - x_2 \geq 1 \\ -x_1 + 3x_2 \geq 5 \end{array}\right\},$$

$$x_1 \geq 0,\ x_2 \geq 0;$$

18. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = 4x_1 + 2x_2 \to \max$$

$$\left.\begin{array}{l} -3x_1 + 2x_2 \le 6 \\ x_1 - 4x_2 \le 2 \\ x_1 - x_2 \le 5 \end{array}\right\},$$

$$x_1 \ge 0,\ x_2 \ge 0;$$

19. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = x_1 + x_2 \to \min$$

$$\left.\begin{array}{l} x_1 + x_2 \ge 1 \\ -x_1 + x_2 \ge 2 \\ -x_1 + x_2 \ge -3 \end{array}\right\},$$

$$x_1 \ge 0,\ x_2 \ge 0;$$

20. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = 4x_1 - 4x_2 \to \min$$

$$\left.\begin{array}{l} 2x_1 - 2x_2 + 8 \ge 0 \\ -x_1 + 4x_2 + 10 \ge 0 \\ 2x_1 + 2x_2 - 12 \ge 0 \end{array}\right\},$$

$$x_1 \ge 0,\ x_2 \ge 0;$$

21. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = 4x_1 + 2x_2 \to \max$$

$$\left.\begin{array}{l} x_1 + 3x_2 \le 6 \\ x_1 \le 3 \\ 2x_1 + x_2 \le 10 \end{array}\right\},$$

$$x_1 \ge 0,\ x_2 \ge 0;$$

22. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = -2x_2 - 4x_3 + 2x_5 \to \max$$

$$\left.\begin{aligned} x_1 - 3x_2 + 6x_3 + x_4 + 10x_5 + x_6 &= 29 \\ x_1 + 5x_2 - 5x_3 - x_4 + 2x_5 &= -5 \\ 2x_2 + x_3 + x_4 - 8x_5 - x_6 &= 2 \end{aligned}\right\},$$

$$x_j \geq 0 \quad \left(j = \overline{1,6}\right)$$

23. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = 2x_1 - x_2 + 3x_3 + x_4 \to \max$$

$$\left.\begin{aligned} 2x_1 + x_2 - 3x_3 &= 10 \\ x_1 + x_3 + x_4 &= 7 \\ -3x_1 - 2x_3 + x_5 &= 4 \end{aligned}\right\},$$

$$x_j \geq 0 \quad \left(j = \overline{1,5}\right)$$

24. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = x_1 + 3x_2 + 2x_3 \to \max$$

$$\left.\begin{aligned} 3x_1 - 2x_2 + x_3 &\geq 5 \\ x_1 + x_2 + 2x_3 &\geq 10 \\ -2x_1 + 3x_2 - x_3 &\geq 2 \end{aligned}\right\},$$

$$x_1 \geq 0,\ x_2 \geq 0,\ x_3 \geq 0$$

25. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = 3x_1 + x_2 + x_3 \to \max$$

$$\left.\begin{aligned} 2x_1 - x_2 + x_3 &= 6 \\ x_1 + 2x_2 - x_3 &= 4 \end{aligned}\right\},$$

$$x_1 \geq 0,\ x_2 \geq 0,\ x_3 \geq 0$$

26. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = x_1 + x_2 + x_3 + x_4 \to \min$$

$$\left.\begin{aligned} x_1 - x_2 - 2x_3 + x_4 &\geq 6 \\ -x_1 + x_3 &\leq 2 \\ 2x_2 - 3x_3 + 2x_4 &\leq 8 \end{aligned}\right\},$$

$x_j \geq 0 \quad \left(j = \overline{1,4}\right)$

27. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = x_1 + 8x_2 + 3x_3 - x_4 \to \max$$

$$\left.\begin{array}{l} x_1 + x_2 - 3x_3 - x_4 = -3 \\ x_1 + 3x_4 = 12 \\ x_2 + x_3 = 1 \end{array}\right\},$$

$x_j \geq 0 \quad \left(j = \overline{1,4}\right)$

28. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = 3x_1 + 3x_2 - 4x_3 \to \max$$

$$\left.\begin{array}{l} 2x_1 + x_2 - 3x_3 \geq 18 \\ 4x_1 - 5x_3 \leq 12 \\ 3x_1 - 2x_2 + x_3 \geq 14 \end{array}\right\},$$

$x_1 \geq 0,\ x_2 \geq 0,\ x_3 \geq 0$

29. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = 6x_1 - x_2 + 3x_3 \to \max$$

$$\left.\begin{array}{l} 3x_1 - 7x_2 + 5x_3 \leq 15 \\ 2x_1 + 3x_2 - 4x_3 = 16 \\ 6x_1 + 5x_2 - 8x_3 \leq 12 \end{array}\right\},$$

$x_j \geq 0 \quad \left(j = \overline{1,3}\right)$

30. Келесі есепті қос мағыналы симплекс әдісімен шешу:

$$z = 6x_1 + 8x_2 + x_3 \to \max$$

$$\left.\begin{array}{l} x_1 + x_2 + x_3 \leq 3 \\ x_1 + 2x_2 \leq 4 \end{array}\right\},$$

$x_j \geq 0 \quad \left(j = \overline{1,3}\right)$

4. ТРАНСПОРТ ЕСЕБІ

Сызықтық бағдарламалаудың ең маңызды есептерінің бірі транспорт есебі болып табылады, яғни, жүкті жөнелту пункттерін оларды жеткізу пункттеріне тасымалдау құны минимал болатындай етіп ең тиімді түрде бекіту есебі болып табылады.

Сызықтық бағдарламалау есептерінің бірі бола отырып, транспорт есебі симплекс әдісінің алгоритмімен шығарылуы мүмкін. Бірақ симплекс әдісін тікелей транспорт есебіне қолдану орынсыз болады, себебі симплекс әдісі сызықтық бағдарламалаудың кез келген есебін шешуге арналған әмбебап әдіс бола тұра, ол транспорт есебінің шарттарының ерекшеліктерін ескермейді, және симплекс әдісін мұндай есепті шешу кезінде қолдану тым қолайсыз болады.

4.1. Транспорт есебінің жалпы қойылымы

Сәйкесінше a_1, a_2, …, a_m бірлік мөлшерінде қандай да бір біртекті өнім шығаратын m пункт бар. Бұл өнімді сәйкесінше b_1, b_2, …, b_n мөлшерде тұтынатын n жеткізу пункттеріне жеткізу керек. Өнім бірлігін i-ші өндіріс пунктінен j-ші жеткізу пунктіне тасымалдау құны c_{ij}-ге тең болсын, ал тасымалданатын өнім бірлігінің сәйкес мөлшері x_{ij} ($i = 1, \ldots, m$; $j = 1, \ldots, n$) болсын.

Транспорт есебінің қойылымы (немесе жеткізушілерді тұтынушыларға бекіту есептері) 1.7 параграфта қарастырылған.

Есептің шарттары төмендегі кесте (екі өлшемді матрица) түрінде ықшамдалып жазылады:

58-кесте

b_j / a_i	b_1	b_2	…	…	b_n
a_1	c_{11} x_{11}	c_{12} x_{12}	…	…	c_{1n} x_{1n}
a_2	c_{21} x_{21}	c_{22} x_{22}	…	…	c_{2n} x_{2n}
…	…	…	…	…	…
a_m	c_{m1} x_{m1}	c_{m2} x_{m2}	…	…	c_{mn} x_{mn}

x_{ij} сандарының m × n жиынын, яғни $\| x_{ij} \|$ матрицасын – тасымалдау жоспары деп, ал $\| c_{ij} \|$ матрицасын − транспорт шығындарының матрицасы деп атайды.

Егер x_{ij} сандары келесі шарттарды қанағаттандырса, жоспар тіректі жоспар деп аталады:

$$\left.\begin{aligned} &x_{ij} \geq 0 \ \ (i = 1, \dots, m; \ \ j = 1, \dots, n), \\ &x_{i1} + x_{i2} + \cdots + x_{in} = a_i \ \ (i = 1, \dots, m), \\ &x_{1j} + x_{2j} + \cdots + x_{mj} = b_j \ \ (j = 1, \dots, n). \end{aligned}\right\} \qquad (4.1)$$

мұндағы алғашқы m теңдік әрбір өндіріс пунктінен өндірілген тауардың барлығы сыртқа шығарылатынын білдіреді, ал соңғы n теңдік әрбір тұтыну пункті толығымен қанағаттандырылатынын білдіреді.

Транспорт есебі тіректі жоспарлар ішінен оңтайлысын іздеп табады, яғни ондағы жалпы тасымалдау құны минимал болу қажет:

$$z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \qquad (4.2)$$

Осылайша, транспорт есебі сызықтық бағдарламалау есептеріне жатады және симплекс әдісінің алгоритмімен

шығарылады. Алайда бұл есептің практикалық маңыздылығы жоғары және шектеулерінің (4.1) ерекше болғандығынан:

а) шектеулер теңдеулер түрінде берілген;

б) әрбір белгісіз айнымалы тек екі теңдеуге ғана кіреді;

в) белгісіз айнымалылардың коэффициенттері – бірліктер болады,

Бұл есепті шешу үшін арнайы алгоритмдер жасалған, олар симплекс әдісінің алгоритміне қарағанда айтарлықтай жеңіл. Солардың бірі – төменде қарастырылған потенциалдар әдісі – ол Л.В. Канторовичтің жалпы әдісінің транспорт есебін шешуге арналған құралы болып табылады және оны Л.В.Канторович пен М.К. Гавурин және кейінірек олардан бөлек Данциг ұсынған болатын.

Транспорт есебінің шешілуінің қажетті және жеткілікті шарты – жіберу пункттеріндегі жүктің барлық қорлары мен жеткізу пункттеріндегі осы жүкке деген қажеттіліктердің теңдігі болып табылады:

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j .$$

Кейде бұл теңдік орындалмай қалады, яғни жүк қоры жеткізу пункттеріндегі қажеттіліктен асып кетеді

$$\sum_{i=1}^{m} a_i > \sum_{j=1}^{n} b_j,$$

немесе жеткізу пункттеріндегі қажеттілік жүк қорынан асып кетеді

$$\sum_{i=1}^{m} a_i < \sum_{j=1}^{n} b_j.$$

Бұл жағдайлар үшін минимал транспорт шығындары бар тасымалдау жоспарын құру туралы есеп қойылуы мүмкін.

Тауар қоры қажеттілік деңгейінен асып кеткен жағдайда

$$\sum_{i=1}^{m} a_i > \sum_{j=1}^{n} b_j$$

оңтайлы тасымалдау жоспарын табу үшін жалған (n+1) – ші жеткізу пунктін енгіземіз:

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

және жүкті осы пунктке тасымалдау құнын нөлге тең деп аламыз: $c_{i,n+1} = 0 \ (i = 1, \dots, m)$. Алынған жаңа есеп қарапайым транспорт есебі болып табылады, және мұнда $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n+1} b_j$ теңдігі орындалады.

Жаңа есептің $\|x_{ij}\|_{m,n+1}$ оңтайлы жоспарынан алынған $\|x_{ij}\|_{m,n}$ жоспары бастапқы есептің оңтайлы жоспары екендігін байқау қиын емес, себебі ескі есептің ең жақсы жоспарының болуы жаңа есептің (оңтайлы жоспардан да) ең жақсы жоспарының болуына алып келуші еді, ал бұл мүмкін емес.

Осылайша

$$\sum_{i=1}^{m} a_i < \sum_{j=1}^{n} b_j$$

болған кезде жалған (m+1) – ші пункті енгізіледі, ондағы жүк қоры

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

және тасымалдау құны $c_{m+1,j} = 0 \ (j = 1, \dots, n)$. Осымен бұл есеп қарапайым транспорт есебіне келтіріледі, оның оңтайлы шешімінен жоғарыдағыдай бастапқы есептің оңтайлы шешімін алуға болады.

4.2. Потенциалдар әдісі

Транспорт есебінің оңтайлы жоспарын анықтау үшін бірнеше әдіс жасалған. Ең жиі қолданылатындары – потенциалдар әдісі және дифференциалды ренталар әдісі. Бұл оқулықта потенциалдар әдісі ұсынылған. Транспорт есебінің оңтайлы шешімін потенциалдар әдісімен анықтаудың жалпы принципі СБЕ-ні симплекс әдісімен шешу принципіне ұқсас, атап айтқанда: алдымен транспорт есебінің тіректі шешімі

табылады, сосын оңтайлы шешім алынғанша ол біртіндеп жақсартылады.

Транспорт есебін шешу үшін келесі кестені қолданайық:

59-кесте

c_{11}	…	c_{1j}	…	c_{1n}	a_1
…	…	…	…	…	…
c_{i1}	…	c_{ij}	…	c_{in}	a_i
…	…	…	…	…	…
c_{m1}	…	c_{mj}	…	c_{mn}	a_m
b_1	…	b_j	…	b_n	a_i / b_j

x_{ij} тасымалдау шамалары жоспарды құру барысында (i, j) ұяшықтарының сол жақ төменгі бұрыштарына енгізіледі.

Жиын деп ұяшықтардың еркін жиынын айтамыз, ал тізбек деп келесі түрдегі жиындарды айтамыз:

$(i_1, j_1), (i_1, j_2), (i_2, j_2), (i_2, j_3), \ldots$

немесе

$(i_1, j_1), (i_2, j_1), (i_2, j_2), (i_3, j_2), \ldots,$

мұнда тізбектің көршілес ұяшықтарының әрбір жұбы не бір жолда, не 2-кестенің бір бағанында орналасады, және тізбектің ешқандай үш ұяшығы бір жолда (бағанда) жатпайды.

Егер соңғы ұяшық бірінші ұяшықпен бір бағанда (жолда) жататын болса, яғни тізбек келесі түрге ие болса

$(i_1, j_1), (i_1, j_2), (i_2, j_2), (i_2, j_3), \ldots, (i_k, j_1)$

немесе

$(i_1, j_1), (i_2, j_1), (i_2, j_2), (i_3, j_2), \ldots, (i_1, j_k),$

онда мұндай тұйық тізбекті цикл деп атаймыз.

Төмендегі кестеде (1, 2), (1, 3), (3, 3), (3, 5), (4, 5), (4, 2) циклы бейнеленген.

60-кесте

c_{11}	c_{12}	c_{13}	c_{14}	c_{15}
c_{21}	c_{22}	c_{23}	c_{24}	c_{25}
c_{31}	c_{32}	c_{33}	c_{34}	c_{35}
c_{41}	c_{42}	c_{43}	c_{44}	c_{45}

Егер бұл жоспардың нөлден өзгеше x_{ij} элементтері бар ұяшықтар жиынында бір де бір цикл болмаса, онда кез келген $X = \|x_{ij}\|_{m,n}$ мүмкін тасымалдау жоспары ациклдық жоспар деп аталады. Кейінірек көретініміздей, оңтайлы тасымалдау жоспарын ациклдық жоспарлар арасынан іздеу жеткілікті және кез келген ациклдық жоспардың $x_{ij} > 0$ элементтерінің N саны келесі теңсіздіктерді қанағаттандырады:

$\max\{m, n\} \le N \le m + n - 1.$

Егер $X = \|x_{ij}\|_{m,n}$ ациклдық жоспарында

$N = m + n - 1$

теңдігі болса, онда $x_{ij} > 0$ элементтері бар (i, j) ұяшықтарының m+n–1 жиынындағы c_{ij} элементтері X-таңдалған деп аталады.

Енді $N < m + n - 1$ болсын. Онда $x_{ij} > 0$ элементтері бар (i, j) ұяшықтар жиынын тағы $x_{ij} = 0$ элементтері бар m+n–1–N ұяшықтарымен толықтырайық, ондағы $m + n - 1$ ұяшықтарының бүкіл жиынында циклдар болмау керек, және осы жаңа жиындағы c_{ij} элементтері X-таңдалған болсын.

Транспорт есебінің қандай да бір $X' = \|x'_{ij}\|_{m,n}$ мүмкін жоспары оңтайлы болуы үшін оған төмендегі шарттарды қанағаттандыратын $u'_1, u'_2, \dots, u'_m; v'_1, v'_2, \dots, v'_n$ m + n сандарынан құралған жүйенің сәйкес келуі қажетті және жеткілікті:

$$v_j - u_i \le c_{ij} \quad (i = 1, \dots, m;\ j = 1, \dots, n) \qquad (4.3)$$

және

$$v_j - u_i = c_{ij} \text{ барлық } x'_{ij} > 0 \ (x'_{ij} \in X') \text{ үшін} \qquad (4.4)$$

u_i', v_j' сандары сәйкесінше жіберу және жеткізу пункттерінің потенциалдары деп аталады, олардың бүкіл жүйесі – потенциалды деп, ал (4.3)-(4.4) шарттары – $\{u_i', v_j'\}$ жүйесінің потенциалдық шарттары деп аталады; жеке алынған әрбір теңсіздік (теңдік) сәйкес (i, j) ұяшығы үшін потенциалдық шарты деп аталады.

Егер $\{u_i', v_j'\}$ потенциалдық шарты бар X жоспарын да потенциалды жоспар деп атасақ, онда теорема қысқаша түрде былай тұжырымдалады: транспорт есебінің жоспары оңтайлы болуы үшін оның потенциалды болуы қажетті және жеткілікті.

Потенциалдар әдісінің алгоритмі алдын ала қадамнан және қайталанатын жалпы қадамнан тұрады.

Алдын ала қадам:

1) Солтүстік-батыс бұрыш әдісімен немесе минималды элемент әдісімен немесе Фогель әдісімен бастапқы X ациклдық жоспары (есептің тіректі шешімі) құрылады.

2) Алынған жоспар үшін $u_1, \dots, u_m;\ v_1, \dots, v_n$ сандарының m + n жүйесі құрылады, мұнда X-таңдалған барлық c_{ij} үшін $v_j - u_i = c_{ij}$ шарттары орындалуы тиіс.

3) Құрылған $u_1, \dots, u_m;\ v_1, \dots, v_n$ жүйесі потенциалдылыққа зерттеледі (яғни X жоспары оңтайлылыққа зерттеледі).

Жалпы қадам (алдыңғы қадамда құрылған X жоспары оңтайлы емес болғанда, яғни $u_1, \dots, u_m;\ v_1, \dots, v_n$ жүйесі потенциалды емес болғанда қолданылады) келесі үш операциядан тұрады:

1) жоспарды жақсарту, яғни X жоспарын жаңа X' жоспарына ауыстыру, жаңа жоспардың құны X жоспарындағы тасымалдау құнынан аспауы тиіс;

2) барлық X-таңдалған c_{ij} үшін $v_j' - u_i' = c_{ij}$ шартын қанағаттандыратын жаңа $u_1', \dots, u_m'; v_1', \dots, v_n'$ жүйесімен X' жоспарын құру ($u_1, \dots, u_m;\ v_1, \dots, v_n$ жүйесін түзету арқылы);

3) құрылған $u'_1, \ldots, u'_m; v'_1, \ldots, v'_n$ жүйесін потенциалдыққа зерттеу.

Барлық есептер үшін алдын ала қадам бір рет қана орындалады, ал жалпы қадам – оңтайлы жоспар алынғанша немесе жүйенің потенциалды еместігіне көз жеткізгенімізше итерациялық түрде бірнеше рет орындалады.

Транспорт есебінің тіректі шешімін табу тәсілдерімен 58-кестенің мысалы арқылы танысайық. Бұл кестенің әрбір ұяшығы (ұяшықтардың сол жақ бағаны мен олардың жоғарғы жолын қоспағанда) белгілі бір жектізуші – тұтынушы жұбына сәйкес келеді. Мысалы, і –ші вертикаль және j – ші горизонталь қатарда орналасқан ұяшық і – ші жеткізуші және j – ші тұтынушы жұбына сәйкес келеді. Ұяшықтарға тиісті бағдар бойынша тасымалдау көлемдері жазылады. Тіректі (мүмкін) шешім (1,1) бағдары бойынша тасымалдау көлемін орнатудан бастап құрылады, яғни кестенің жоғарғы сол жақ («солтүстік-батыс») ұяшығын толтырудан басталады. Оны есеп шарттары бойынша максимал мүмкін деп қабылдайық, яғни

$X_{11}=\min(a_1,b_1)$.

Егер $a_1<b_1$ болса, онда 1-ші жеткізуші өз мүмкіндіктерін толығымен пайдаланды деген сөз, және қалған тасымалдарды орнату кезінде оны ескермесе де болады, ал 1-ші тұтынушының қажеттілігі b_1 - a_1 болады. Егер $a_1>b_1$ болса, онда 1-ші тұтынушы өзінің өнімге деген қажеттілігін толығымен қанағаттандырды деген сөз, және оны ары қарай ескермесе де болады, ал 1-ші жеткізушіде өнімнің тек a_1-b_1 бірлігі ғана қалды. Егер $a_1=b_1$ болса, онда ары қарай жеткізушіні де тұтынушыны да қарастырмаса болады. Алайда мұндай жағдайда бұлардың біреуі ғана «ойыннан шығып қалды» деп санаймыз (мысалы, жеткізуші), ал тұтынушының мүмкін болатын қажеттіліктері нөлге тең деп санаймыз.

Осы тұжырымдардан шығатыны – (1,1) бағдар бойынша тасымал көлемдерін орнатқаннан кейін, бізде жаңа есеп пайда болады, ондағы жеткізушілер мен тұтынушылар санының жалпы саны бастапқы есептегіден 1-ге кем. Осы жаңа есеп ескі есеп ұяшықтарының алғашқы горизонталь немесе вертикаль қатарын ойша сызып тастау арқылы алынады, осы есептегі

солтүстік-батыс ұяшығына тасымалдың максимал мүмкін көлемін жазамыз. Осы процесті жалғастыра отырып, тіректі шешімді аламыз, себебі

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n+1} b_j = d.$$

Есептің тіректі шешімін құрудың бұл тәсілі солтүстік-батыс бұрыш әдісі деп аталады.

Тіректі шешімді солтүстік-батыс бұрыш әдісімен құру кезінде өнімді тасымалдаудың c_{ij} құны еш ескерілмейді. Алайда бастапқы тіректі шешімде арзан тасымалдар көп болған сайын бұл шешім оңтайлы шешімге жақындай түседі. Сондықтан бастапқы кестеден, сосын одан кейінгі қарастырылатын бөліктерінен ең арзан бағдарға сәйкес келетін ұяшықты таңдап, оған максимал мүмкін тасымалды жазу келісілді. Тіректі шешімді құрудың мұндай жолы минималды элемент әдісі деп аталады. Кестенің қарастырылып отырған бөлігінің бірінші жолының немесе керісінше, бірінші бағанының минималды c_{ij} элементін таңдауға болады. Бұл жерде (кестеде, жолда немесе бағанда) бірдей мәнді c_{ij} минималды элементтері бірнешеу болатын жағдайлар кездеседі. Сондықтан олардың бірін қандай ережемен толтыру керектігін алдын-ала келісіп алу қажет. Мысалы, мұндай жағдайларда бірінші индексі ең кіші ұяшықты таңдауға, ал егер бірінші индекстері бірдей болып қалса, екінші индексі ең кіші ұяшықты таңдауға келісіп алуға болады. Бұл реттеудің лексикографиялық ережесі деп аталады. Тіректі шешімді құру ережесінің жоғарыда көрсетілген түрлендірулерін қолдану кезінде, солтүстік-батыс ұяшықты таңдау кезіндегідей, әрбір қадамда тек қана жолды (немесе тек қана бағанды) алып тастап отырған жөн. Егер іс жүзінде жол да, баған да шығып қалатын болса, онда формальды түрде нөлдік жеткізілімі бар жеткізуші (немесе сәйкесінше нөлдік қажеттілігі бар тұтынушы) енгізіледі. Бұл әдіс транспорт есебінің солтүстік-батыс бұрыш әдісінің көмегімен табылған жоспар кезіндегі жалпы тасымал құнынан төмен тасымал құны бар тіректі жоспарын табуға мүмкіндік беретінін айта кеткен жөн.

Транспорт есебінің тікректі шешімін табудың келесі әдісі Фогель әдісі болып табылады. Фогель әдісі бойынша барлық бағандар мен барлық жолдар бойынша әрбір итерацияда оларда

жазылған минимал тарифтердің арасындағы айырма табылады. Бұл айырмалар есеп шарттарының кестесіндегі солар үшін арнайы тағайындалған жол мен бағанға жазылады. Көрсетілген айырмалардың ішінен минималдысы таңдалады. Берілген айырма сәйкес келетін жолда (немесе бағанда) минималды тасымалдау құны анықталады. Ол жазылған ұяшық берілген итерацияда толтырылады. Егер минимал баға берілген бағанның (жолдың) бірнеше ұяшығында бірдей болса, онда осы бағанда (жолда) орналасқан екі минимал тарифтің арасындағы ең үлкен айырмаға сәйкес келетін бағанда (жолда) орналасқан ұяшық таңдалады.

9-мысал.

Потенциалдар әдісінің көмегімен келесі транспорт есебін шығару керек:

$a_1 = 25$, $a_2 = 55$, $a_3 = 20$;

$b_1 = 45$, $b_2 = 15$, $b_3 = 20$, $b_4 = 20$;

$$c = \begin{pmatrix} 9 & 5 & 3 & 10 \\ 6 & 3 & 8 & 2 \\ 3 & 8 & 4 & 8 \end{pmatrix}$$

Ш е ш у і . Келесі кестені құрамыз

61-кесте

	b_j			
a_i	~~45~~ ~~20~~ 0	~~15~~ 0	~~20~~ 0	~~20~~ 0
~~25~~ 0	9 25	5 0	3 0	10 0
~~55~~ ~~35~~ 20	6 20	3 15	8 20	2 0
~~20~~ 0	3 0	8 0	4 0	8 20

Алдын ала қадам.

1. Бастапқы (тіректі) жоспарды құру. Солтүстік-батыс бұрыш әдісімен табамыз:

$x_{11} = \min\{a_1, b_1\} = \min\{25, 45\} = 25$, $x_{12} = x_{13} = x_{14} = 0$.

Қорлар бағанына және тұтыну жолына өзгерістер енгіземіз.

Енді $x_{21} = \min\{55, 20\} = 20$, $x_{31} = 0$.

Бұл жерде бірінші жеткізу пунктінің қажеттілігі толығымен қанағаттандырылды да, ол нөлге айналады, ал екінші жіберу пунктінің қоры 55 – 20 = 35.

Ары қарай: $x_{22} = \min\{35, 15\} = 15$, $x_{32} = 0$.

Осылайша $x_{23} = \min\{20, 20\} = 20$, $x_{33} = x_{24} = 0$.

Және, $x_{34} = \min\{20, 20\} = 20$.

$Z_{тірек} = 25*9 + 20*6 + 15*3 + 20*8 + 20*8 = 225 + 120 + 45 + 160 + 160 = 710$.

2. Барлық X-таңдалған c_{ij} үшін u_i, v_j жүйесін құру. Бос емес ұяшықтар саны m + n – 1 = 3 + 4 – 1 = 6-дан аз болғандықтан, ең кіші тарифі бар бос ұяшыққа «базистік» нөлді жазамыз да, осы ұяшықты бос емес деп санаймыз. u_i және v_j сандарын келесі сызықтық теңдеулерден анықтаймыз.

$v_1 - u_1 = 9$,

$v_1 - u_2 = 6$,

$v_2 - u_2 = 3$,

$v_3 - u_2 = 8$,

$v_4 - u_3 = 8$.

$u_1 = 0$ деп алып, $v_1 = 9$, $u_2 = 3$, $v_2 = 6$, $v_3 = 11$, $v_4 = 5$, $u_3 = -3$ табамыз.

Оларды кестеге жазайық:

62-кесте

	v_j			
u_i	9	6	11	8
0	9 25 -	5	3 +	10
3	6 20 +	3 15	8 - 20	2
0	3	8	4	8 20

3. Құрылған u_i, v_j жүйесін потенциалдыққа зерттеу.

Егер Х-таңдалған болып табылмайтын барлық c_{ij} үшін келесі теңсіздіктер орындалмаса,

$v_j - u_i \le c_{ij}$, онда u_i, v_j жүйесі потенциалды. Осылайша,

$v_2 - u_1 = 6 - 0 = 6 < 5$? Жоқ, демек $\alpha_{12} = 6 - 5 = 1$,

$v_3 - u_1 = 11 - 0 = 11 < 3$? Жоқ, демек $\alpha_{13} = 11 - 3 = 8$,

$v_4 - u_1 = 8 - 0 = 8 < 10$? Иә,

$v_1 - u_3 = 9 - 0 = 9 < 3$? Жоқ, демек $\alpha_{31} = 9 - 3 = 6$,

$v_2 - u_3 = 6 - 0 = 6 < 8$? Иә,

$v_3 - u_3 = 11 - 0 = 11 < 4$? Жоқ, демек $\alpha_{33} = 11 - 4 = 7$.

Демек, берілген жүйе потенциалды емес.

Шарттар орындалмайтын теңсіздіктер үшін мыналар есептеледі:

$\alpha_{ij} = v_j - u_i - c_{ij}$.

Барлық α_{ij} арасынан ең үлкен мәнді таңдаймыз, яғни $\alpha_{i_0 j_0} = \max(\alpha_{ij}) = \max(1,8,6,7) = 8$.

Жалпы қадамға көшеміз.

Жалпы қадам.

1. Жоспарды жақсарту. Ары қарай $\theta = \min(X_{ij}^-) = \min(25,20) = 20$ табамыз. Теріс таңбалы жартылай тізбек ұяшықтарында x_{ij} мәні θ-ге кемиді, ал оң таңбалы жартылай тізбек ұяшықтарында x_{ij} мәні θ-ге артады.

63-кесте

	v_j			
u_i	9	6	4	5
0	9 – 5	5	3 + 20	10
3	6 40	3 15	8	2 20
0	3 +	8	4 – 20	8 20

2. Барлық X-таңдалған c_{ij} үшін жаңа u_i, v_j жүйесін құру. u_i және v_j сандарын сызықтық теңдеулер жүйесінен анықтаймыз.

$v_1 - u_1 = 9$,
$v_3 - u_1 = 3$,
$v_1 - u_2 = 6$,
$v_2 - u_2 = 3$,
$v_4 - u_2 = 2$,
$v_3 - u_3 = 8$.

$u_1 = u_3 = 0$ деп алып, $v_1 = 9$, $u_2 = 3$, $v_2 = 6$, $v_3 = 3$, $v_4 = 5$ табамыз.

3. Құрылған u_i, v_j жүйесін потенциалдыққа зерттеу.

$v_2 - u_1 = 6 - 0 = 6 > 5$? Жоқ, демек $\alpha_{12} = 6 - 5 = 1$,
$v_3 - u_1 = 4 - 0 = 4 > 3$? Жоқ, демек $\alpha_{13} = 4 - 3 = 1$,
$v_4 - u_1 = 5 - 0 = 5 < 10$? Иә,
$v_2 - u_2 = 6 - 3 = 3 < 3$? Иә,
$v_3 - u_2 = 4 - 3 = 1 < 8$? Иә,
$v_4 - u_2 = 8 - 3 = 5 < 2$? Жоқ, демек $\alpha_{24} = 5 - 2 = 3$,
$v_1 - u_3 = 9 - 0 = 9 > 3$? Жоқ, демек $\alpha_{31} = 9 - 3 = 6$,
$v_2 - u_3 = 6 - 0 = 6 < 8$? Иә,
$v_4 - u_3 = 5 - 0 = 5 < 8$? Иә.

Жүйе потенциалды емес болғандықтан, α-ң ең үлкен мәнін табамыз, яғни $\alpha_{i_o j_o} = \max(\alpha_{ij}) = \max(1,1,6) = 6$ және (3, 1) ұяшықтан бастап цикл құрамыз. Келесі жақсартылған жоспарды аламыз.

64-кесте

u_i \ v_j	9	3	8	2
0	9 25 –	5	3 +	10
0	6 +	3 15	8 – 20	2 20
6	3 20	8	4	8

Жаңа потенциалдар жүйесін құрамыз,

$v_1 - u_1 = 9$,

$v_2 - u_2 = 3$,

$v_3 - u_2 = 8$,

$v_4 - u_2 = 2$,

$v_1 - u_3 = 3$.

$u_1 = u_2 = 0$ деп алып, $v_1 = 9$, $u_3 = 6$, $v_2 = 3$, $v_3 = 8$, $v_4 = 2$ табамыз.

3. Құрылған u_i, v_j жүйесін потенциалдыққа зерттеу.

$v_2 - u_1 = 3 - 0 = 3 < 5$? Иә,

$v_3 - u_1 = 8 - 0 = 8 > 3$? Жоқ, демек $\alpha_{13} = 8 - 3 = 5$,

$v_4 - u_1 = 2 - 0 = 2 < 10$? Иә,

$v_1 - u_2 = 9 - 0 = 9 > 6$? Жоқ, демек $\alpha_{21} = 9 - 6 = 3$,

$v_2 - u_3 = 3 - 6 = -3 < 8$? Иә,

$v_3 - u_3 = 8 - 6 = 2 < 4$? Иә,

$v_4 - u_3 = 2 - 6 = -4 < 8$? Иә.

Жүйе әлі де потенциалды емес болғандықтан, α-ң ең үлкен мәнін табамыз, яғни $\alpha_{i_0 j_0} = \max(\alpha_{ij}) = \max(5,3) = 5$ және (1, 3) ұяшықтан бастап цикл құрамыз. Келесі жақсартылған жоспарды аламыз.

65-кесте

	v_j			
u_i	9	6	3	5
0	9 5 –	5 +	3 20	10
3	6 20 +	3 – 15	8	2 20
6	3 20	8	4	8

Келесі потенциалдар жүйесін аламыз,

$v_1 - u_1 = 9$,
$v_3 - u_1 = 3$,
$v_1 - u_2 = 6$,
$v_2 - u_2 = 3$,
$v_4 - u_2 = 2$,
$v_1 - u_3 = 3$.

$u_1 = 0$ деп алып, $v_1 = 9$, $u_2 = 3$, $v_2 = 6$, $v_3 = 3$, $v_4 = 5$, $u_3 = 6$ табамыз, оны потенциалдыққа зерттейміз

$v_2 - u_1 = 6 - 0 = 6 > 5$? Жоқ, демек $\alpha_{12} = 6 - 5 = 1$,
$v_4 - u_1 = 5 - 0 = 5 < 10$? Иә,
$v_3 - u_2 = 3 - 3 = 0 < 8$? Иә,
$v_2 - u_3 = 6 - 6 = 0 < 8$? Иә,
$v_3 - u_3 = 3 - 6 = -3 < 4$? Иә,
$v_4 - u_3 = 5 - 6 = -1 < 8$? Иә.

Циклды (1, 2) ұяшықтан бастаймыз да, келесі жоспарды аламыз

66-кесте

	v_j			
u_i	9	6	3	5
0	9	5 5	3 20	10
3	6 25	3 10	8	2 20
6	3 20	8	4	8

Потенциалдар жүйесін құрамыз,

$v_2 - u_1 = 5$,

$v_3 - u_1 = 3$,

$v_1 - u_2 = 6$,

$v_2 - u_2 = 3$,

$v_4 - u_2 = 2$,

$v_1 - u_3 = 3$.

$u_1 = 0$ деп алып, $v_2 = 5$, $v_3 = 3$, $u_2 = 3$, $v_1 = 9$, $v_4 = 5$, $u_3 = 6$ табамыз да, оны потенциалдыққа зерттейміз

$v_1 - u_1 = 9 - 0 = 9 = 9$? Иә.

$v_4 - u_1 = 5 - 0 = 5 < 10$? Иә.

$v_3 - u_2 = 3 - 3 = 0 < 8$? Иә.

$v_2 - u_3 = 6 - 6 = 0 < 8$? Иә.

$v_3 - u_3 = 3 - 6 = -3 < 4$? Иә.

$v_4 - u_3 = 5 - 6 = -1 < 8$? Иә.

Алынған жүйе потенциалды болғандықтан, оңтайлы жоспар алынды және $z_{опт} = 5*5 + 20*3 + 25*6 + 10*3 + 20*2 + 20* 3 = 25 + 60 + 150 + 30 + 40 + 60 = 365$.

Ең қарапайым транспорт есебін қарастыруды қорытындылай келе, потениалдар әдісінің бұл есепті шешудің жалғыз әдісі еместігін айта кету керек. Олардың кейбірі бір жағдайларда тиімдірек болады, ал басқалары басқа жағдайларда тиімдірек болады. (4.1)-(4.2) түріндегі модельге мазмұны бойынша транспортқа, тасымалды жоспарлауға еш қатысы жоқ

есептер келтірілуі мүмкін. Мұндай жағдайларда берілген есеп транспорт есебінің терминінде тұжырымдала алады деп айтады. Ең қарапайым транспорт есебін шешуге арналған тиімді алгоритмдер бар болғандықтан, есепті әрқашан да транспорт есебінің терминінде тұжырымдауға тырысу керек, әрине егер ол алдын-ала мүмкін емес болмаса.

Тасымалды жоспарлаумен байланысты практикалық есептердің көпшілігі қарастырылған есептің аясына кірмейді, себебі қандай да бір қосышма шектеулермен күрделенген (мысалы, транспорт магистралінің өткізгіштік қабілетінің шектеулігі).

(4.1)-(4.2) есеп жүктің өңделуі жүзеге асырылатын аралық транспорт түйіндерінің болуымен күрделенуі мүмкін (мысалы, басқа транспорт түріне ауыткумен), немесе жеке тұтынушылар мен жеткізушілер арасында транспорт магистральдарының болмауымен күрделенуі мүмкін.

Көбінесе бір емес, бірнеше өнім түрлерін тасымалдаудың оңтайлы жоспарын құру керек болады.

Көптеген практикалық есептер қарапайым транспорт есебі жеке жағдай болып табылатын модельге келтіріледі. Транспорт есебіндегідей, мұндағы шектеулер екі топқа бөлінеді, ондағы әрбір айнымалы әрбір топтың тек бір шектеуіне ғана кіреді, яғни транспорт есебінің ерекшеліктерінің бірі сақталады. Алайда, транспорт есебімен салыстырғанда, шектеуге айнымалы кірген жағдайда шектеулер тобының бірінде міндетті түрде бірлік немесе нөл емес коэффициенттерге емес, еркін коэффициенттерге рұқсат беріледі.

Мынадай мысалды қарастырайық. Энергетикалық отынның әр түрлі түрін өндіретін m өндіріс пункті бар: $p_1,p_2,\ldots,p_m$, олардың өндіріс көлемі сәйкесінше $a_1,a_2,\ldots,a_m$ салмақ бірліктері, және осы отынды тұтынатын n тұтынушы бар, олардың қажеттіліктері, мысалы, оларды қамтамасыз ететін жылу калорияларымен өрнектелген, $b_1,b_2,\ldots,b_n$. j-ші тұтынушы жағдайында қолданылған і-ші түрлі отынның бір салмақ бірлігі a_{ij} жылу калориясын қамтамасыз етеді; отынның бір салмақ бірлігін p_i пунктінен j-ші тұтынушыға жеткізуге кететін транспорт шығындары c_{ij} ақша бірлігіне сәйкес келеді. Жалпы

транспорт шығындары минимал болатындай етіп, отынды тұтынушылар арасында үлестіру керек.

Транспорт есебінің 1.7-параграфта келтірілген математикалық моделіне ұқсатып, берілген есептің математикалық моделі келесідей болады:

1. Жалпы транспорт шығындарын минималдау:

$$z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \rightarrow \min \qquad (4.5)$$

2. Өндіріс бойынша шектеулер:

$$\sum_{j=1}^{n} x_{ij} \leq a_i, \quad i = \overline{1,m} \qquad (4.6)$$

3. Қажеттілік бойынша шектеулер:

$$\sum_{i=1}^{m} a_{ij} x_{ij} = b_j, \quad j = \overline{1,n} \qquad (4.7)$$

4. Мазмұнына байланысты:

$$x_{ij} \geq 0, \quad i = \overline{1,m}, \quad j = \overline{1,n}. \qquad (4.8)$$

(4.5)-(4.8) түріндегі есептер үлестіру есептері деп аталады. Олар үшін түрлі шешу алгоритмдері жасалған, олар транспорт есебіне арналған алгоритмдерге қарағанда күрделірек болып келеді, бірақ сызықтық бағдарламалаудың жалпы әдістерімен салыстырғанда айтарлықтай жеңіл. Олардың кейбіреулері үшін арнайы шешу әдістері бар. Транспорт типті есептердің көпшілігі оларға сәйкес тораптардың (графтардың) қасиеттерін математикалық талдауға негізделген әдістермен шешіледі, яғни транспорт түйіндері мен коммуникацияларын шартты графикалық бейнелеу арқылы шешіледі.

Транспорт (тек транспорт қана емес) есептерін шешудің тораптық әдістері оларды матрицалық қойылымда шешу әдістеріне қарағанда әлдеқайда тиімді болып келеді.

Оқулықтың осы бөлімінде транспорт есебінің жеке жағдайын қарастырайық, ол тағайындаулар туралы есеп немесе таңдау есебі деп аталады.

4.3. Венгр әдісі

Тағайындаулар туралы есепті шешудің арнайы әдісін қарастырайық. Тағайындаулар туралы есептің мысалы 1.7-параграфта келтірілген.

$$X_{ij} = \begin{cases} 1, \text{ егер i-ұшақ j-авиалинияға белгіленген болса,} \\ 0, \text{ егер i-ұшақ j-авиалинияға белгіленбеген болса.} \end{cases}$$

Жоғарыда берілген шарт тағайындаулар туралы есепті сызықтық бағдарламалау есептерінің класынан шығарып тастайды, себебі олар сызықтық емес. Айнымалыларына осындай шектеулер қойылған математикалық бағдарламалау есептері бульдік айнымалылары бар есептер деп аталады. Қалған барлық есептері мен мақсатты функциясы сызықтық түрде болғандықтан, формальды түрде бұл есепті бульдік айнымалылары бар сызықтық бағдарламалау есебіне жатқызуға болады. Практикалық тұрғыда бұл есепті транспорт есебінің жеке жағдайы ретінде қарастыруға болады. Шын мәнісінде, соңғы шарттарды айнымалылардың теріс еместік шартымен алмастыра отырып алып тастайтын болсақ, есебіміз қарапайым транспорт есебіне айналады, оның транспорт есебінен бір ғана айырмашылығы болады: ондағы барлық a_i мен b_j-лар 1-ге тең. Егер осы есепті потенциалдар әдісімен немесе a_i мен b_j бүтін болғанда бүтін санды оңтайлы шешімге алып келетін кез келген басқа әдіспен шығаратын болсақ, онда алынған шешім біз ескермей кеткен айнымалылардың бульдік шартын автоматты түрде қанағаттандыратын болады: егер бірнеше бүтін сандық теріс емес айнымалылардың қосындысы 1-ге тең болса, онда әрбір айнымалы тек 1-ге немесе 0-ге тең бола алады.

$$C=\begin{pmatrix} c_{11}\ c_{12}\ \ldots c_{1n} \\ c_{21}\ \ c_{22}\ldots c_{2n} \\ \ldots\ldots\ldots\ldots\ldots\ldots \\ c_{m1}\ c_{m2}\ldots c_{mn} \end{pmatrix}$$

- тағайындаулар туралы есептің тиімділік (өнімділік) матрицасы болсын. Осы есептің қойылымына сәйкес әрбір жолдан және әрбір бағаннан бір-бірден c_n элементтерін таңдау арқылы шешу керек (жол тағайындалатын объекттерге сәйкес келеді, мысалы, ұшақтарға, ал бағандар тағайындау жасалатын объекттерге сәйкес келеді, мысалы әуе линияларына), ондағы таңдалған элементтердің қосындысы жалпы тиімділікке тең, және ол ең үлкен болуы керек. Осы есепті әрбір жолы мен бағанында нөлдері бар қандай да бір теріс емес элементтері бар матрицасынан әрбір бағаны мен әрбір жолында бір-бірден n нөлдік элементтерді таңдауға қалай келтіруге болатындығын қарастырайық. Ол үшін келесі анықтама мен теореманы қолданайық.

Екі $C=(c_{ij})$ және $D=(d_{ij})$ матрицалары эквивалентті деп аталады, егер олардың біреуі екіншісінің әрбір жолының элементтеріне бір санды қосу арқылы алынатын болса (әр түрлі жолдар үшін бұл сандар әр түрлі болуы мүмкін) және әрбір бағанының элементтеріне бір санды қосу арқылы алынатын болса (әр түрлі бағандар үшін бұл сандар әр түрлі болуы мүмкін). Онда екінші матрица да біріншісінің жолдары мен бағандарына тұрақты сандарды қосу арқылы алынатыны анық, сондықтан бұл қасиет екеуіне бірдей.

Мысалы,

$$C=\begin{pmatrix} 4 & 0 & -1 & 5 \\ 2 & 3 & 0 & -1 \\ 1 & 1 & 2 & 3 \\ -1 & -2 & -3 & 0 \end{pmatrix} \qquad D=\begin{pmatrix} 6 & 3 & 3 & 6 \\ 4 & 6 & 4 & 0 \\ 4 & 5 & 7 & 5 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

матрицалары эквивалентті, себебі D матрицасы C матрицасының жолдарына 0, 0, 1, -1 сандарын, ал бағандарына сәйкесінше 2, 3, 4, 1 сандарын қосу арқылы алынады.

Теорема 3. Эквивалентті матрицалары бар екі таңдау есебінің оңтайлы тағайындауларының жиыны сәйкес келеді [15].

Матрицалардың эквиваленттілік қасиеті бір-біріне сәйкес болғандықтан, D матрицасы бар есептің кез келген оңтайлы таңдауы C матрицасы бар есептің оңтайлы таңдауы болып табылатыны рас.

Сонымен, C матрицасының келесі екі тізбекті түрлендіруін жасайық:

$$C=(c_{ij})\rightarrow C_1=(c_{ij}')=(\max c_{ij}-c_{ij}) \rightarrow D=(d_{ij})=(c_{ij}'-\min c_{ij}') \quad (4.9)$$

Бұл түрлендірулер алдын-ала түрлендірулер деп аталады. Егер C матрицасын бірінші рет түрлендіргеннен кейін алынған C_1 матрицасының әрбір жолында ең болмаса бір нөл бар болса, онда екінші түрлендіру, әрине, C матрицасын өзгертпейді және ол орындалмайды да.

Теріс емес матрицаның n элементтерінің қосындысының ең кіші мүмкін мәні нөлге тең екені анық. Осылайша, біздің есеп D матрицасынан, немесе теріс емес элементтері бар оған эквивалентті матрицадан әр жол мен әр бағаннан бір-бірден n нөлдік элементтерді таңдауға келтіріледі. Төменде келтірілген алгоритмнің бейресми мағынасы: нөлдерді толық емес дұрыс таңдаудан басқа таңдауға біртіндеп көшу болып табылады, бұл таңдауда алдыңғысына қарағанда бір нөл артық болады, және толық дұрыс таңдау алынғанша осылай жалғаса береді. Бұл жерде бөлек кезеңдерде алдыңғысына эквивалентті болып табылатын жаңа матрицаға көшу керек болуы мүмкін.

Венгр әдісінің алгоритмі:

Берілген есептің C тиімділік матрицасының алдын-ала түрлендірілуі жасалып қойған болсын дейік, және әрбір жолы мен әрбір бағанында ең болмаса бір нөлдік элементі бар теріс емес D матрицасы алынған болсын.

1^0. D матрицасының бірінші бағанындағы қандай да бір нөлді жұлдызшамен белгілейік (0*). Екінші бағаннан, бірақ бірінші бағандағы 0* тұрған жолда жатпайтын қандай да бір нөлді жұлдызшамен белгілейік (егер мұндай нөл екінші

бағаннан табылса). Үшінші бағаннан әлі 0* жоқ жолда жатқан қандай да бір нөлді жұлдызшамен белгілейік (егер мұндай нөл үшінші бағаннан табылса). Бұл процедураны матрицаның барлық бағандарынан өтіп болғанша жалғастырамыз.

Егер жұлдызшамен белгіленген нөлдер саны n болса, онда процес аяқталды: жұлдызшалары бар нөлдер тұрған орындар бастапқы есептің оңтайлы шешімінде мәндері 1-ге тең n x_{ij} айнымалыларына сәйкес келеді.

Егер жұлдызшалары бар нөлдер n-нен аз болса, онда 2^0 қадамға көшеміз.

2^0. Матрицадағы 0* бар бағандардың жоғарғы жағына плюс ("+") таңбасын қоямыз да, осы бағандарды бос емес деп санаймыз.

Процес барысында бос емес жолдар да пайда болады. Бос емес баған мен бос емес жолдың қиылысында тұрған элементтер бос емес болып саналады, ал қалғандары – бос.

Егер матрицада бос емес нөлдер болмаса, онда 5^0 қадамға көшеміз.

Егер бос емес нөлдер бар болса, (матрица жолдарын рет-ретімен солдан оңға қарай қарай отырып) олардың біріншісін таңдаймыз. Оны штрихпен белгілеп аламыз (0'). Егер оның жолында жұлдызшалы нөл болмаса, онда 4^0 қадамға көшеміз. Егер оның жолында 0* бар болса, онда 3^0 қадамды орындаймыз.

3^0. Жаңа ғана штрихпен белгіленген нөлі бар жолда жатқан 0*-ң бағанын босатамыз ("+" белгісін алып тастап, оны қайтадан бос деп санаймыз). Біздің 0' тұрған жолдың оң жағына "+" таңбасын қоямыз да, оны бос емес деп санаймыз. 2^0 қадамға көшеміз.

4^0. Жаңа ғана белгіленген 0'-тен бастап, нөлдердің тізбегін құрамыз: осы 0'-ден баған бойынша 0*-ға және мүмкін болғанша т.б. Тізбек қандай да бір 0'-де (бірінші 0'-де де болуы мүмкін) үзіліп қалады. Тізбектегі нөлдердің жұлдызшаларын алып тастаймыз да, тізбектегі нөлдердің штрихтарын жұлдызшаға ауыстырамыз. Жұлдызшалы нөлдердің жаңа жиынында алдыңғысына қарағанда бір нөл артық болады және ол да дұрыс болып табылады.

Жұлдызшалардан басқа барлық белгілерді алып тастаймыз да, 1^0 қадамға көшеміз.

5^0. Матрицаның бос элементтерінің арасынан минимал элементті табамыз (ол h-ке тең болсын), сосын оны барлық бос жолдардан азайтамыз да, барлық бос емес бағандарға қосамыз. Бұл жерде ешқандай белігеулерді алып тастамаймыз. Алдыңғы матрицаға эквивалентті болып табылатын және бос нөлдері бар матрица алынады. 2^0 қадамға көшеміз.

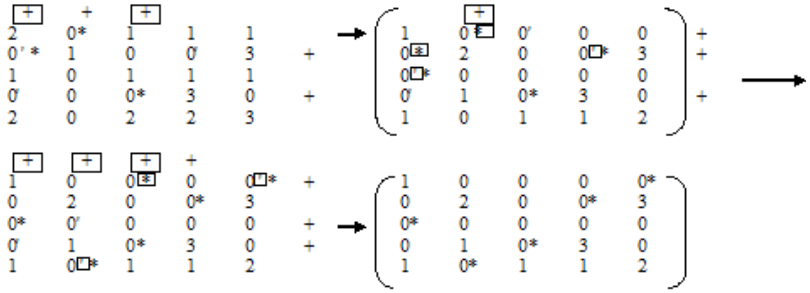
10-мысал. Тиімділік матрицасы келесідей берілген болса, тағайындаудың оңтайлы нұсқасын табу керек:

$$\begin{pmatrix} 2 & 3 & 3 & 5 & 4 \\ 4 & 2 & 4 & 6 & 2 \\ 2 & 2 & 2 & 4 & 3 \\ 4 & 3 & 4 & 3 & 5 \\ 0 & 1 & 0 & 2 & 0 \end{pmatrix}$$

Есепті шешу процесінде алынатын, тиісті белгілеулері бар матрицалар тізбегін келтірейік. Белгілердің алып тасталуы олардың тік бұрышқа алынуымен белгіленді. Матрицадан көшу бағыттарының үстінде сәйкес түрлендірулер кезінде қолданылған алгоритмнің пункттері көрсетілген.

2	3	3	5	4
4	2	4	6	2
2	2	2	4	3
4	3	4	3	5
0	1	0	2	0

2	0	1	1	1
0	1	0	0	3
2	1	2	2	2
0	0	0	3	0
4	2	4	4	5



Түрлендіру процесі аяқталды, себебі жұлдызшалы нөлдердің саны n=5 болды. Тағайындаулардың оңтайлы нұсқасы: $x_{15}= x_{24}= x_{31} = x_{43} = x_{52} = 1$,

Қалған x_{ij} =0, яғни бірінші ұшақ бесінші әуе жолына, екіншісі – төртіншіге, үшіншісі – біріншіге, төртіншісі – үшіншіге, бесіншісі – екіншіге тағайындалады.

Сұрақтар мен тапсырмалар

1. Потенциалдар әдісі транспорт есебінің қандай модельдеріне қолданылады?
2. Тіректі жоспарды құрудың қандай әдістері бар?
3. Транспорт есебінің шешілуіне қандай шарт қажетті әрі жеткілікті шарт болып табылады?
4. Транспорт есебінің тіректі шешімін анықтаудың қай әдісінде транспорт шығындары ескеріледі?
5. Жүйені потенциалдыққа зерттеу шарттарын атаңыз.
6. Қандай жоспар ациклдық жоспар деп аталады?
7. Транспорт есебінің математикалық моделінің ерекшеліктерін атап шығыңыз.
8. Жабық транспорт есебін шығарыңыз:

$a_1 = 40$, $a_2 = 50$, $a_3 = 60$;

$b_1 = 35$, $b_2 = 40$; $b_3 = 40$, $b_4 = 35$;

$$c = \begin{pmatrix} 4 & 9 & 7 & 3 \\ 2 & 4 & 1 & 5 \\ 7 & 6 & 3 & 1 \end{pmatrix};$$

9. Ашық транспорт есебін шығарыңыз:

$a_1 = 160$, $a_2 = 140$, $a_3 = 60$;

$b_1 = 80$, $b_2 = 80$; $b_3 = 60$, $b_4 = 80$;

$$c = \begin{pmatrix} 5 & 4 & 3 & 4 \\ 3 & 2 & 5 & 5 \\ 1 & 6 & 3 & 2 \end{pmatrix};$$

10. Фирма үш кәсіпорынды біріктіреді, олардың әрқайсысы үш түрлі бұйым шығарады. Бір бұйымды дайындау кезіндегі әрбір кәсіпорынның өнімділігі (ақша бірлігімен) келесі кестемен сипатталады:

	Бұйымдар		
Кәсіпорындар	15	6	12
	6	9	13
	8	11	2

Әрбір кәсіпорынның тек бір бұйымға ғана мамандандырылу қажеттілігін ескере отырып, бұйымдардың өндірісін кәсіпорындарға бөліп беру керек, осы үлестіруде фирманың жалпы өнімділігі максимал болуы тиіс.

11. Қандай есептер транспорт есебінің терминдерінде тұжырымдала алады?

12. Тағайындаулар туралы есепті шешу әдістерін атаңыз.

13. Транспорт есебін шығарыңыз:

$a_1 = 30$, $a_2 = 40$, $a_3 = 20$;

$b_1 = 20$, $b_2 = 25$, $b_3 = 35$, $b_4 = 10$;

$$c = \begin{pmatrix} 2 & 3 & 5 & 4 \\ 3 & 2 & 4 & 1 \\ 4 & 3 & 2 & 6 \end{pmatrix};$$

14. Транспорт есебін шығарыңыз:

$a_1 = 115$, $a_2 = 70$, $a_3 = 68$;

$b_1 = 95$, $b_2 = 38$, $b_3 = 50$, $b_4 = 70$;

$$c = \begin{pmatrix} 4 & 7 & 3 & 9 \\ 2 & 1 & 8 & 5 \\ 7 & 9 & 6 & 1 \end{pmatrix};$$

15. Транспорт есебін шығарыңыз:
$a_1 = 60$, $a_2 = 70$, $a_3 = 40$;
$b_1 = 20$, $b_2 = 30$; $b_3 = 30$, $b_4 = 50$;

$$c = \begin{pmatrix} 2 & 4 & 5 & 1 \\ 2 & 3 & 9 & 4 \\ 3 & 4 & 2 & 5 \end{pmatrix};$$

16. Транспорт есебін шығарыңыз:
$a_1 = 30$, $a_2 = 60$, $a_3 = 10$;
$b_1 = 15$, $b_2 = 40$, $b_3 = 25$, $b_4 = 20$;

$$c = \begin{pmatrix} 7 & 3 & 6 & 4 \\ 2 & 5 & 3 & 9 \\ 8 & 1 & 7 & 3 \end{pmatrix};$$

17. Транспорт есебін шығарыңыз:
$a_1 = 100$, $a_2 = 150$, $a_3 = 80$;
$b_1 = 80$, $b_2 = 140$, $b_3 = 110$;

$$c = \begin{pmatrix} 4 & 3 & 5 \\ 10 & 1 & 2 \\ 3 & 8 & 6 \end{pmatrix};$$

18. Транспорт есебін шығарыңыз:
$a_1 = 4$, $a_2 = 6$, $a_3 = 10$;
$b_1 = 3$, $b_2 = 5$, $b_3 = 12$;

$$c = \begin{pmatrix} 1 & 0 & 2 \\ 3 & 5 & 4 \\ 1 & 2 & 3 \end{pmatrix};$$

19. Транспорт есебін шығарыңыз:
$a_1 = 12$, $a_2 = 14$, $a_3 = 4$;
$b_1 = 9$, $b_2 = 10$, $b_3 = 11$;

$$c = \begin{pmatrix} 5 & 1 & 8 \\ 2 & 4 & 0 \\ 3 & 6 & 7 \end{pmatrix};$$

20. Транспорт есебін шығарыңыз:
$a_1 = 40$, $a_2 = 35$, $a_3 = 45$;
$b_1 = 35$, $b_2 = 35$; $b_3 = 25$, $b_4 = 30$;

$$c = \begin{pmatrix} 6 & 5 & 4 & 6 \\ 1 & 6 & 7 & 10 \\ 4 & 2 & 9 & 9 \end{pmatrix};$$

21. Транспорт есебін шығарыңыз:
$a_1 = 40$, $a_2 = 45$, $a_3 = 70$;
$b_1 = 48$, $b_2 = 30$; $b_3 = 29$; $b_4 = 40$;

$$c = \begin{pmatrix} 3 & 6 & 4 & 3 \\ 2 & 3 & 1 & 3 \\ 6 & 5 & 1 & 4 \end{pmatrix};$$

22. Транспорт есебін шығарыңыз:
$a_1 = 36$, $a_2 = 40$, $a_3 = 60$;
$b_1 = 50$, $b_2 = 42$; $b_3 = 31$, $b_4 = 41$;

$$c = \begin{pmatrix} 6 & 9 & 5 & 8 \\ 7 & 4 & 9 & 7 \\ 3 & 5 & 7 & 5 \end{pmatrix};$$

23. Транспорт есебін шығарыңыз:
$a_1 = 220$, $a_2 = 175$, $a_3 = 130$;
$b_1 = 70$, $b_2 = 115$; $b_3 = 40$, $b_4 = 30$,b_5=60;

$$c = \begin{pmatrix} 4 & 5 & 2 & 8 & 6 \\ 3 & 1 & 9 & 7 & 3 \\ 9 & 6 & 7 & 2 & 1 \end{pmatrix};$$

24. Транспорт есебін шығарыңыз:
$a_1 = 85$, $a_2 = 150$, $a_3 = 50$;
$b_1 = 75$, $b_2 = 80$; $b_3 = 60$, $b_4 = 100$;

$$c = \begin{pmatrix} 6 & 7 & 3 & 5 \\ 1 & 2 & 5 & 6 \\ 8 & 10 & 20 & 1 \end{pmatrix};$$

25. Транспорт есебін шығарыңыз:
$a_1 = 120$, $a_2 = 220$, $a_3 = 100$;
$b_1 = 180$, $b_2 = 80$; $b_3 = 160$, $b_4 = 90$,b_5=50;

$$c = \begin{pmatrix} 1 & 2 & 3 & 1 & 4 \\ 6 & 3 & 4 & 5 & 2 \\ 8 & 2 & 1 & 9 & 3 \end{pmatrix};$$

26. Транспорт есебін шығарыңыз:
$a_1 = 80$, $a_2 = 190$, $a_3 = 90$;
$b_1 = 110$, $b_2 = 60$; $b_3 = 170$, $b_4 = 80$;

$$c = \begin{pmatrix} 8 & 1 & 9 & 7 \\ 4 & 6 & 2 & 12 \\ 3 & 4 & 8 & 9 \end{pmatrix};$$

27. Транспорт есебін шығарыңыз:
$a_1 = 30$, $a_2 = 30$, $a_3 = 10$;
$b_1 = 50$, $b_2 = 30$; $b_3 = 10$, $b_4 = 20$;

$$c = \begin{pmatrix} 1 & 2 & 4 & 1 \\ 2 & 3 & 1 & 5 \\ 3 & 2 & 4 & 4 \end{pmatrix};$$

28. Транспорт есебін шығарыңыз:
$a_1 = 150$, $a_2 = 170$, $a_3 = 110$;
$b_1 = 110$, $b_2 = 120$; $b_3 = 80$, $b_4 = 50$; $b_5 = 70$;

$$c = \begin{pmatrix} 7 & 2 & 11 & 5 & 9 \\ 8 & 4 & 3 & 6 & 1 \\ 3 & 5 & 10 & 7 & 8 \end{pmatrix};$$

29. Транспорт есебін шығарыңыз:
$a_1 = 40$, $a_2 = 70$;
$b_1 = 30$, $b_2 = 60$, $b_3 = 20$;

$$c = \begin{pmatrix} 10 & 6 & 8 \\ 5 & 4 & 7 \end{pmatrix};$$

30. Транспорт есебін шығарыңыз:
$a_1 = 40$, $a_2 = 60$;
$b_1 = 20$, $b_2 = 70$, $b_3 = 10$;

$$c = \begin{pmatrix} 6 & 4 & 2 \\ 3 & 5 & 7 \end{pmatrix}$$

5. БҮТІН САНДЫҚ БАҒДАРЛАМАЛАУ

Бұл тарауда барлық айнымалыларға (толықтай бүтін сандық есептер) немесе айнымалылардың бір бөлігіне (жартылай бүтін сандық есептер) қойылатын қосалқы шарттары бар сызықтық бағдарламалау есептері қарастырылады.

Айнымалыларға қойылатын бүтін сандық шартын ұсынатын себептерге байланысты бүтін сандық бағдарламалау есептерінің келесідей екі моделі ажыратылады: бөлінбейтін есептер және альтернативтік айнымалылары бар есептер. Бірінші модельге айнымалылары бөлінбейтін шамаларды өрнектейтін есептер моделі жатады, мысалы, үлестірілетін машиналар саны, кәсіпорындар саны, бөлінбейтін жүк бірлігінің саны, дайындалатын өнім саны және т.б. Мұндай есептер моделінің 2-тарауда қарастырылған қарапайым сызықтық бағдарламалау есептерінен еш айырмашылығы жоқ. Бүтін сандық шарты есеп мазмұнын мұқият талдау негізінде белгіленеді. Бөлінбейтін есептердің ерекше түрі бүтін сандық транспорт есептері болып табылады, бұл есептерде бастапқы деректер бүтін болған жағдайда оның шешімі де автоматты түрде бүтін болады.

Альтернативтік айнымалылары бар модельдер комбинаторлық сипаттағы әр түрлі оңтайлы есептерді, бейсызықтық бағдарламалау есептерін, қосымша логикалық шарттары бар есептерді (мысалы «немесе-немесе», «егер-онда» және т.б.) қамтиды, бұл есептер жалған альтернативтік айнымалыларды (яғни «0» немесе «1» мәндерін қабылдайтын айнымалыларды) енгізу арқылы бүтін сандық бағдарламалау есептерінің сызықтық модельдеріне келтіріледі.

Бүтін сандық бағдарламалау есептерінің шешілуі арнайы құрылған СБЕ-лердің ақырғы тізбегінің шешілуіне келтіріледі: $z_1,\ldots,z_s,\ldots,z_n$, бұлардың әрқайсысы алдыңғысының шарттарына қосымша сызықтық шектеу (теңсіздік) қою арқылы алынады, ол «қима» деп аталады.

Сондай-ақ s-ші қима деп z_{s-1} есебіне енгізілетін сызықтық шектеу айтылады, ол z_s есебін құру үшін енгізіледі және келесі екі шартты қанағаттандырады:

1) z_{s-1} есебінің кез келген бүтін шешімі оны қанағаттандырады;

2) z_{s-1} есебінің кез келген бүтін емес шешімі оны қанағаттандырмайды ("кесіп түсіріледі").

Бүтін сандық бағдарламалау әдістері қиманы қалыптастыру тәсіліне байланысты ажыратылады.

Толық немесе жартылай бүтін сандық есептердің шешілуі тізбекті итерациялар түрінде орындалады, олардың әрқайсысы келесі пункттерді қамтиды:

1. z_{s-1} есебі шешіледі де, оның x^*_{s-1} оңтайлы шешімі табылады;

2. Егер x^*_{s-1} шешімі бүтіндік шарттарын қанағаттандырса, онда есептеу процесі аяқталады, ал егер бұл шарттар қанағаттандырылмаса, 3-пунктке көшеміз;

3. z_{s-1} есебін шешу кезінде алынған соңғы симплекс кестесінің негізінде қосымша шектеу (қима) жазылады;

4. Алдыңғы пунктте табылған шектеуді z_{s-1} есебіне қосу нәтижесінде z_s есебі алынады, одан кейін қайтадан 1-пунктке барамыз.

Тарихта алғашқы бүтін типті есеп венгр математигі Е.Эгерваридің 1932 жылы жарық көрген персоналды тағайындау туралы есебі болып табылады [16], ол жоғарыда (1.7-параграфты қараңыз) айтылған әуе линияларында ұшақтарды тағайындау туралы есепке ұқсас тұжырымдалады.

Бүтін сандық бағдарламалау саласындағы жүйелі зерттеулер 1955 жылы басталды, ол кезде сызықтық бағдарламалау бойынша Екінші симпозиумда бомбалаушы туралы есеп қарастырылған (осы оқулықтың 6 және 7-тарауларын қараңыз).

Барлық айнымалыларға немесе айнымалылардың бір бөлігіне бүтіндік талабы қойылатын есептер тізімі айтарлықтай үлкен. Айнымалыларға бүтін сандық талабы өндірісті орналастыру есептерінде, өндірістік бағдарламаны орындау үшін машиналар паркін сатып алу туралы есептерде, маусымдық жұмысшылар санын анықтау есептерінде және т.б. есептерде қойылады.

Есептің қойылымы. Әрбір j-шісі (j=1,2,…, n) a_{ij} көрсеткішімен және c_j пайдалылығымен і-ші сипаттамаға ие болатын бөлінбейтін заттардың n жиынының ішінен b_i (i=1,2,…,m) шамасындағы ресурстарды қолданудың тиімділігін максималдауға мүмкіндік беретін жиынды табу керек.

Бұл есептің математикалық моделі келесі түрде берілуі мүмкін:

$$Z = \sum_{j=1}^{n} c_j x_j \to \max (\min) \tag{5.1}$$

$$\left.\begin{array}{l} \sum_{j=1}^{n} a_{ij} x_i \le b_i,\ i = \overline{1,m} \\ x_j \ge 0,\ j = \overline{1,n} \end{array}\right\} \tag{5.2}$$

$$x_{j\ -\ бүтін},\ j=\overline{1,n} \tag{5.3}$$

Егер $n_1=n$ болса, онда (5.1)-(5.3) бүтін сандық бағдарламалау есебінің моделі болып табылады, егер $n_1<n$ болса, онда жартылай бүтін сандық бағдарламалау есебінің моделі болып табылады.

Бүтін сандық бағдарламалау есебінің жеке жағдайы бульдік айнымалылары бар есеп болып табылады. Оның математикалық моделі жалпы түрде келесідей жазылады:

мақсатты функцияны максималдау:

$$Z = \sum_{j=1}^{n} c_j x_j \to \max (\min) \tag{5.4}$$

шектеулері:

$$\sum_{j=1}^{n} a_{ij} x_i \le b_i,\ i = \overline{1,m}. \tag{5.5}$$

$$x_j = \begin{cases} 1,\ j=1,2,...,n \\ 0,\ \text{кері жағдайда} \end{cases} \tag{5.6}$$

Барлық айнымалыларына немесе айнымалыларының бір бөлігіне қойылатын бүтін сандық шарты дискреттілік талабына

ауыстырылған есептер бүтін сандық бағдарламалау есептерінің класына жанасады. Атап айтқанда, әрбір j-ші x_j (j=1,2,…,n) айнымалысы үшін ол қабылдай алатын мәндер жиыны (бүтін болуы міндетті емес) алдын-ала анықталған:

$x_j \in A_j$, мұндағы $A_j=\{0, A_{j2},\ldots, A_{jkj}\}$.

A_{jkj} ранжирленген деп жорамалданады, яғни $A_{jk1}< A_{jk2}<\ldots < A_{jkj}$.

Дискреттік бағдарламалаудың жалпы есебінің математикалық моделі келесі түрде жазылуы мүмкін:

$$Z = \sum_{j=1}^{n} c_j x_j \to \max(\min) \qquad (5.7)$$

$$\sum_{j=1}^{n} a_{ij} x_i \le b_i, \quad i = \overline{1,m}, \qquad (5.8)$$

$$x_j \in A_j = \{0, A_{j2},\ldots, A_{jkj}\}, j=1,2,\ldots,n_1, n_1 \le n. \qquad (5.9)$$

(5.9) шарты бұл есептер класының атын анықтады. Егер $n_1=n$ болса, онда (5.7)-(5.9) есебі дискреттік бағдарламалау есебі деп аталады, ал егер $n_1<n$ болса, онда (5.7)-(5.9) есебі жартылай дискреттік бағдарламалау есебі деп аталады.

(5.1)-(5.3) есебінің (5.3) шарттары мен (5.4)-(5.6) есебінің (5.6) шарты (5.7)-(5.9) есебінің (5.9) шартының жеке жағдайы екендігін байқау қиын емес. Шынымен де, (5.3) шарты j=1,2,…,n; l=1,2,…; $A_{jkj}=\infty$ үшін $A_{jl} =A_{jl-1}+1$ болғандағы жағдайға сәйкес келеді. (5.9) шарты $A_j=\{0;1\}$, j=1,2,…,n болғандағы жағдайға сәйкес келеді.

Бүтін типті есептер үшін тіректі және оңтайлы шешім ұғымы анықталған.

(5.1)-(5.3) шарттарын (сәйкесінше (5.7)-(5.9) шарттарын) қанағаттандыратын $x=(x_1,x_2,..,x_n)$ векторы (5.1)-(5.3) есебінің (сәйкесінше (5.7)-(5.9) есебінің) тіректі шешімі деп аталады. (5.1) функциясы (сәйкесінше (5.7) функциясы) ең үлкен (ең кіші) мәнге ие болатын тіректі шешім оңтайлы шешім деп аталады.

Тіректі және оңтайлы шешім ұғымдарын анықтай отырып, оларды табу туралы сұрақ туындайды. Бүтін санды есепті шешудің қарапайым жолы тиісті сызықтық есепті шешіп, оның оңтайлы жоспарын ең жақын бүтін санға дейін дөңгелектеу

сияқты көрінеді. Бірақ шын мәнісінде мұндай тәсіл көп жағдайда оңтайлы шешімнен алыстатып қана қоймай, кейде есептің мүмкін емес шешіміне алып келеді.

Бүтін сандық бағдарламалау есептерінің арасында ізделінетін экстремалды шешімі сандардың белгілі (мүмкін болатын) жиынын орын ауыстыру (комбинациялау) арқылы сипатталатын есептер үлкен қызығушылық тудыруда. Бұл есептер комбинаторлы типті есептер деп аталады.

Мұндай типті есептерге (персоналды) тағайындау туралы есеп жатады, ол 1, 2,…,n сандарының (p_1, p_2,…,p_n) орын ауыстыруы арқылы шешіледі. Әрбір тағайындау сәйкестікпен сипатталады: p_i i-ге сәйкес келеді (i=1,2,…,n). Алайда тағайындау туралы есеп сызықтық бағдарламалау есебі түрінде жазыла алады, және ол алдыңғы бөлімде қарастырылған транспорт есебінің жеке жағдайы болып табылады, және шешу әдісі бойынша күрделі болып табылмайды. Транспорт есептерін шешу әдістері бүтін шешімдерді беретіні белгілі. Өтілген есептерге тоқталмастан, оларды шешудің әлі қарастырылмаған модельдері мен әдістерін үйренейік.

5.1. Бүтін сандық бағдарламалау есебінің графикалық түсіндірілуі

Бүтін сандық сызықтық бағдарламалау есебінің математикалық моделі келесі түрде берілсін:

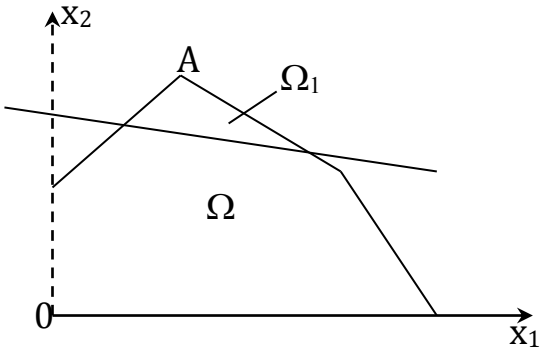
$$Z = \sum_{j=1}^{n} c_j x_j \to \max \tag{5.10}$$

$$\left.\begin{array}{l} \sum_{i=1}^{m} a_{ij} x_i \le b_i, \ \ j = \overline{1,n} \\ x_j \ge 0, \ j = \overline{1,n} \end{array}\right\} \tag{5.11}$$

$$x_{j\ -\ \text{бүтін},}\ \overline{j=1,n} \tag{5.12}$$

Жазықтықта тік бұрышты декарттық координаталар жүйесін қарастырайық (14-сурет). Геометриялық тұрғыда (5.10)-

(5.12) есебінің шешімі тіректі шешімдердің (5.11) шектеулерімен анықталатын бүкіл облысы емес (Ω), жеке дискретті нүктелер болып табылады:



14-сурет

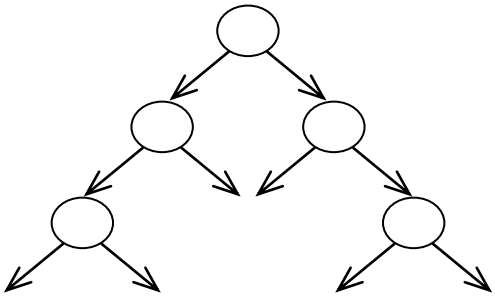
Бүтін сандық сызықтық бағдарламалау есебінің оңтайлы шешімін табудың екі тәсілі белгілі:

1) кесіп түсіру әдістері (мысалы, Гоморидің кесіп түсіру әдісі);

2) комбинаторлы әдістер (мысалы, бұтақтар мен шекаралар әдісі).

Кесіп түсіру әдістерінің мәні мынада: (5.10)-(5.12) есебінің шешімі қадамдардың ақырғы санымен ізделінеді. Әрбір қадамда (5.10)-(5.11) үздіксіз есебі шешіледі, яғни бүтінтік шартынсыз сызықтық бағдарламалау есебі шешіледі, және әрбір қадамда қосымша шектеу енгізіледі. Әрбір қадамда енгізілетін қосымша шектеулер (5.11) шектеулерімен анықталатын Ω шешімдер облысын біртіндеп кішірейтеді. Сондай-ақ енгізілетін әрбір шектеу Ω облысының ең болмағанда бір бүтін нүктесінен өте отырып, оның қандай да бір бөлігін кесіп түсіреді және ол Ω облысынан бүтін нүктелерді алып тастамайды.

«Бұтақтар мен шекаралар» әдісінің мәні тіректі шешімдер жиынын реттелген қиылыспайтын ішкі жиындарға біртіндеп бөлуде болып табылады, олар көбінесе қандай да бір графтың – нұсқалар ағашының төбелері ретінде қарастырылады.



15-сурет

Әрбір төбесі үшін тиісті релаксталған есепті шешу арқылы мақсатты функцияның жоғарғы $(Z \to \max)$ немесе төменгі $(Z \to \min)$ бағасы есептеледі. Релаксация дегеніміз кейбір шектеулерден, мысалы бүтіндік шарттарынан бас тарту болып табылады. Рекорд мәннен нашар бағасы бар төбе алып тасталады. Қалған қандай да бір төбе нұсқалардың ішкі жиынына бөлінеді, олар жаңа төбелерді құрады. Барлық төбелер қаралып, алып тасталмайынша процес жалғаса береді.

5.2. Гоморидің кесіп түсіру әдісі

1) Алгоритмнің бастапқы қадамы.

(5.10)-(5.11) есебі тіке симплекс әдісімен шығарылады. Қадамдардың ақырғы санынан кейін 67-кестені аламыз, ондағы барлық бос мүшелер $(b_j, j = \overline{1,n})$ мен Z-жол коэффициенттері $(q_j, j = \overline{1,n})$ оң таңбалы болып табылады. Егер 67-кестедегі $b_1,\ldots,b_s$ бос мүшелерінің мәндері бүтін болса, онда (5.10)-(5.12) есебінің шешімі табылды деген сөз:

67-кесте

	$-y_1$	…	$-y_s$	$-x_{s+1}$	…	$-y_n$	1
x_1	b_{11}	…	b_{1s}	$b_{1,s+1}$	…	b_{1n}	b_1
…	…	…	…	…	…	…	…
x_s	b_{s1}	…	b_{ss}	$b_{s,s+1}$	…	b_{sn}	b_s
y_{s+1}	$b_{s+1,1}$	…	$b_{s+1,s}$	$b_{s+1,s+1}$	…	$b_{s+1,n}$	b_{s+1}
…	…	…	…	…	…	…	…
y_m	b_{m1}	…	b_{ms}	$b_{m,s+1}$	…	b_{mn}	b_m
Z	q_1	…	q_s	q_{s+1}	…	q_n	Q

$b_1,\ldots,b_m \geq 0$,
$q_1,\ldots,q_n \geq 0$,
$X^*_1 = b_1$,
… … …
$X^*_s = b_s$,
$Z^* = Q$.

Егер X^*_j мәндері бүтін болса, онда есептеу тоқтатылады. Егер X_j айнымалыларының мәндері бүтін емес болса, онда жалпы қадамға көшеміз.

2) Алгоритмнің жалпы қадамы.
Ыңғайлылық үшін 67-кестені келесі түрде жазайық:

68-кесте

	$-\xi_1$	$-\xi_2$	…	$-\xi_n$	1
n_1	b_{11}	b_{12}	…	b_{1n}	b_1
n_2	b_{21}	b_{22}	…	b_{2n}	b_2
…	…	…	…	…	…
n_m	b_{m1}	b_{m2}	…	b_{mn}	b_m
Z	q_1	q_s	…	q_n	Q

Қосымша шектеу келесі түрде құрылады

$$S_i = \sum_{j=1}^{n}(-\beta_{ij})(-\xi_j) - \beta_i \geq 0, \qquad (5.13)$$

мұндағы $\beta_{ij}=b_{ij}-[b_{ij}]$, $\beta_i=b_i-[b_i]$,
$[b_{ij}]$, $[b_i]$ $-b_{ij}$ және b_i сандарының бүтін бөліктері.

(5.13) қосымша шектеуі мүмкін шешім облысынан бүтін шешімді сақтай отырып, бүтін емес оңтайлы шешімі бар бөлікті кесіп түсіреді.

69-кесте

	$-\xi_1$	$-\xi_2$	…	$-\xi_n$	1
n_1	b_{11}	b_{12}	…	b_{1n}	b_1
n_2	b_{21}	b_{22}	…	b_{2n}	b_2
…	…	…	…	…	…
n_m	b_{m1}	b_{m2}	…	b_{mn}	b_m
S_i	$-\beta_{i1}$	$-\beta_{i1}$	…	$-\beta_{in}$	$-\beta_i$
Z	q_1	q_s	…	q_n	Q

68-кестеге (5.13) шектеуін енгізген соң есептің шешуін (69-кесте) қос мағыналы симплекс әдісімен жалғастырамыз.

11-мысал.

$Z=3x_1+7x_2\rightarrow max$

$2x_1+x_2\leq 25$

$x_1+2x_2\leq 5$

$x_1\geq 0$, $x_2\geq 0$

x_1, x_2 – бүтін.

70-кесте

	$-x_1$	$-x_2$	1
y_1	2	1	25
y_2	1	[2]	5
Z	-3	-7	0

71-кесте

	$-x_1$	$-y_2$	1
y_1	3/2	-1/2	45/2
x_2	1/2	1/2	5/2
Z	1/2	7/2	35/2

Оңтайлы шешімі: $x_1=0$, $x_2=5/2$, алайда x_2 айнымалысының мәні бүтін емес. S_2 қосымша шектеуі енгізіледі:

$$S_2 = -\frac{1}{2}x_1 - \frac{1}{2}y_2 - \frac{1}{2} \geq 0$$

72-кесте

	$-x_1$	$-y_2$	1
y_1	3/2	-1/2	45/2
x_2	1/2	1/2	5/2
S_2	[-1/2]	-1/2	-1/2
Z	1/2	7/2	35/2

Есептеуді қос мағыналы симплекс әдісімен жалғастырамыз.

73-кесте

	$-S_2$	$-y_2$	1
y_1	-3/2	1	-42/2
x_2	-1/2	0	1
x_1	1	-1/2	-1/2
Z	-1/2	-6/4	-34/4

74-кесте

	$-S_2$	$-y_2$	1
y_1	3	-2	21
x_2	1	0	2
x_1	-2	1	1
Z	1	3	17

11-мысалдың оңтайлы бүтін мәні:
$Z^*=17$; $X^*\{1, 2\}$.

5.3. Ленд-Дойг әдісі

Ленд-Дойг әдісі толық және жартылай бүтін сандық есептер үшін қолданылады.

Ленд-Дойг алгоритмі:

1^0. (5.10)-(5.11) сызықтық бағдарламалау есебі шешіледі.

2^0. Тексеру: x_j айнымалыларының мәндері бүтін бе? Егер бүтін болса, онда (5.10)-(5.12) есебінің оңтайлы шешімі табылды деген сөз. Егер бүтін болмаса, онда 3^0 қадамға көшеміз.

3^0. Сызықтық бағдарламалаудың екі есебі құрылып, шығарылады:

1-есеп	2-есеп
$Z = \sum_{j=1}^{n} c_j x_j \to \max$ (5.14)	$Z = \sum_{j=1}^{n} c_j x_j \to \max$ (5.17)
$\sum_{i=1}^{n} a_{ij} x_j \le b_j, j = \overline{1,n},$ (5.15)	$\sum_{i=1}^{n} a_{ij} x_j \le b_j, j = \overline{1,n},$ (5.18)
$x_j \ge 0, j = \overline{1,n}$	$x_j \ge 0, j = \overline{1,n}$
$x_j \le [x_j]$ (5.16)	$x_j \le [x_j] + 1.$ (5.19)

Егер қайсыбір есептің (бірінші немесе екінші есептің) шешімі болмаса, онда сәйкес мақсатты функцияға ең кішкентай мән меншіктеледі, яғни "$-\infty$".

4^0. Табылған шешімдердің арасынан мақсатты функцияның мәні ең үлкені таңдалады да, 2^0 қадамға көшеміз.

12-мысал.

$Z=3x_1+4x_2 \to \max$

$3x_1+2x_2 \le 8$

$x_1+4x_2 \le 10$

$x_1 \ge 0, x_2 \ge 0$

x_1, x_2 – бүтін.

Оңтайлы шешімі бүтін емес:

75-кесте

	$-y_1$	$-y_2$	1
x_1	4/10	-2/10	12/10
x_2	-1/10	3/10	22/10
Z	8/10	6/10	124/10

$x_1=12/10$, $x_2=22/10$ – бүтін емес.

Екі есеп құрастырылады:

1-есеп:

$Z=3x_1+4x_2\rightarrow max$

$3x_1+2x_2\leq 8$

$x_1+4x_2\leq 10$

$x_1\leq 1$

$x_1\geq 0$, $x_2\geq 0$.

2-есеп:

$Z=3x_1+4x_2\rightarrow max$

$3x_1+2x_2\leq 8$

$x_1+4x_2\leq 10$

$x_1\geq 2$

$x_1\geq 0$, $x_2\geq 0$.

1-есеп шығарылады:

76-кесте

	$-x_1$	$-x_2$	1
y_1	3	2	8
y_2	1	4	10
y_3	1	0	1
Z	-3	-4	0

77-кесте

	$-x_1$	$-y_2$	1
y_1	10	-2	12
x_2	1	1	10
y_3	4	0	4
Z	-8	4	40

78-кесте

	$-x_1$	$-y_2$	1
y_1	10/4	-2/4	3/4
x_2	1/4	1/4	10/4
y_3	1	0	1
Z	-2	1	10

79-кесте

	$-y_3$	$-y_2$	1
y_1	-10/4	…	1/2
x_2	-1/4	…	9/4
x_1	1	0	1
Z	2	1	12

1-есептің шешімі: $\{x_1=1, x_2=9/4, z_1=12\}$.

2-есеп шығарылады:

80-кесте

	$-x_1$	$-x_2$	1
y_1	3	2	8
y_2	1	4	10
y_4	-1	0	-2
Z	-3	-4	0

81-кесте

	$-y_4$	$-y_2$	1
y_1	-3	-2	-2
y_2	-1	-4	-8
x_1	1	0	-2
Z	3	4	-6

82-кесте

	$-y_4$	$-x_2$	1
y_1	3	2	2
y_2	1	4	8
x_1	-1	0	2
Z	-3	-4	6

83-кесте

	$-y_4$	$-y_1$	1
x_2	3	1	2
y_2	-10	-4	8
x_1	-2	0	4
Z	6	4	20

84-кесте

	$-y_4$	$-y_1$	1
x_2	3/2	1/2	1
y_2	-5/2	-2	4
x_1	-1	0	2
Z	3	2	10

2-есептің шешімі:
$\{x^*_1=2, x^*_2=1, z^*_2=10\}$.

1-есепті шешу кезінде x_2 айнымалысының мәні бүтін емес болды. 1-есеп үшін тағы екі есеп құрылады:

3-есеп:
$Z=3x_1+4x_2\rightarrow max$
$3x_1+2x_2\leq 8$
$x_1+4x_2\leq 10$
$x_1\leq 1$
$x_2\leq 2$
$x_1\geq 0, x_2\geq 0.$

4-есеп:
$Z=3x_1+4x_2\rightarrow max$
$3x_1+2x_2\leq 8$
$x_1+4x_2\leq 10$
$x_1\leq 1$
$x_2\geq 3$
$x_1\geq 0, x_2\geq 0.$

3-есеп шығарылады:

85-кесте

	$-x_1$	$-x_2$	1
y_1	3	2	8
y_2	1	4	10
y_3	1	0	1
y_5	0	1	2
Z	-3	-4	0

86-кесте

	$-x_1$	$-y_5$	1
y_1	3	-2	4
y_2	1	-4	2
y_3	1	0	1
x_2	0	1	2
Z	-3	4	8

87-кесте

	$-y_3$	$-y_5$	1
y_1	-3	…	1
y_2	-1	…	1
x_1	1	0	1
x_2	0	…	2
Z	3	4	11

3-есептің оңтайлы бүтін шешімі:
$\{x^*_1=1, x^*_2=1, z^*_3=11\}$.

4-есеп шығарылады:

88-кесте

	$-x_1$	$-x_2$	1
y_1	3	2	8
y_2	1	4	10
y_3	1	0	1
y_6	0	-1	-3
Z	-3	-4	0

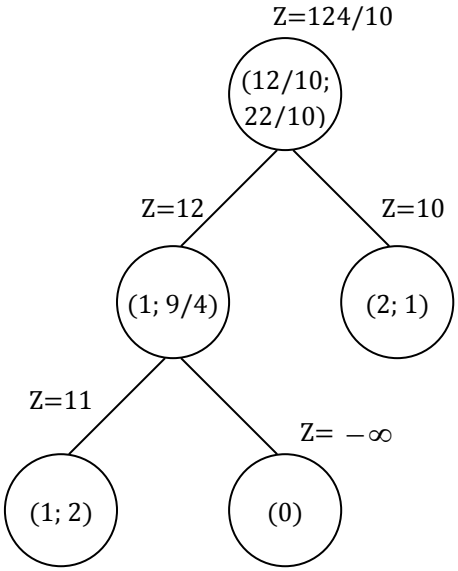
89-кесте

	$-x_1$	$-y_2$	1
y_1	10	-2	12
x_2	1	1	10
y_3	4	0	4
y_6	1	1	-2
Z	-8	4	40

4-есептің шешімі жоқ, демек, $z_4=-\infty$.

Көрнекілік үшін 12-мысалдың төрт есебінен алынған шешімдерді төбелер түрінде нұсқалар ағашында бейнелейік. Нұсқалар ағашынан "$-\infty$" шешімі бар есептерді (4-есеп) және бүтін емес есептерді (1-есеп) сызып тастайық. Сонда нұсқалар ағашында бүтін шешімдері бар екі төбе қалады (2 және 3-есептер), олардың ішіндегі $\{x^*_1=1, x^*_2=1, z^*_3=11\}$ шешімі 2-мысалдың шешімі болып табылады (16-суретті қараңыз).

Нұсқалар ағашы:



16-сурет

Сұрақтар мен тапсырмалар

1. Бүтін сандық бағдарламалау есебінің қойылымын келтіріңіз.
2. Дискретті бағдарламалау есебінің оңтайлы шешімін іздеудің негізгі тәсілдерін атап айтыңыз.
3. «Бұтақтар мен шекаралар» әдісінің мәні неде?
4. Қандай есепті релакстанған есеп деп атайды?
5. Дискретті бағдарламалаудың қай әдісі есептің барлық оңтайлы шешімдерін береді?
6. Ленд-Дойг әдісінің әрбір итерациясында сызықтық бағдарламалаудың қандай екі есебі шешіледі?
7. Ленд-Дойг әдісінің күрделілігі неде?
8. Гомори әдісін қолданып келесі есепті шығарңыз:

$z = 3x_1 + 3x_2 \to \max$

$$\left.\begin{array}{r} x_1 + 3x_2 \geq 6 \\ 3x_1 + 2x_2 \leq 36 \\ x_2 \leq 13 \end{array}\right\},$$

$x_j \geq 0$ – бүтін, $(j = 1,2)$

9. Ленд-Дойг әдісін қолданып келесі есепті шығарыңыз:

$$z = x_1 \to \max$$

$$\left.\begin{array}{r} x_1 + 3x_2 + x_3 = 12 \\ 3x_1 - 8x_2 + x_4 = 24 \end{array}\right\},$$

$$x_j \geq 0 - \text{бүтін}; \left(j = \overline{1,4}\right)$$

10. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = x_1 + 2x_2 \to \max$$

$$\left.\begin{array}{r} 2x_1 + 2x_2 \leq 7 \\ 4x_1 - 5x_2 \leq 9 \end{array}\right\},$$

$x_j \geq 0$ – бүтін, $(j = 1,2)$

11. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = x_1 + x_2 \to \max$$

$$\left.\begin{array}{r} 2x_1 + x_2 \leq 18 \\ x_1 + 2x_2 \leq 16 \end{array}\right\},$$

$x_j \geq 0$ – бүтін, $(j = 1,2)$

12. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = 3x_1 + 4x_2 \to \max$$

$$\left.\begin{array}{r} 3x_1 + 2x_2 \leq 8 \\ x_1 + 4x_2 \leq 10 \end{array}\right\},$$

$x_j \geq 0$ – бүтін, $(j = 1,2)$

13. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = 4x_1 + 3x_2 \to \max$$

$$\left.\begin{array}{r} 2x_1 - x_2 \geq 0 \\ 0{,}5x_1 - x_2 \leq 0 \\ x_1 + x_2 \leq 5 \end{array}\right\},$$

$x_j \geq 0$ – бүтін, $(j = 1{,}2)$;

14. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = -x_2 \to \min$$

$$\left.\begin{array}{l} 3x_1 + 2x_2 \leq 6 \\ -3x_1 + 2x_2 \leq 0 \end{array}\right\}$$

$$x_1,\ x_2 \geq 0$$

$x_1,\ x_2$ – бүтін

15. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = 3x_1 + x_2 \to \min$$

$$\left.\begin{array}{l} -4x_1 + x_2 \leq 29 \\ 3x_1 - x_2 \leq 15 \\ 5x_1 + 2x_2 \geq 38 \end{array}\right\}$$

$$x_1,\ x_2 \geq 0$$

$x_1,\ x_2$ – бүтін

16. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = x_1 + x_2 \to \max$$

$$\left.\begin{array}{r} 3x_1 + 2x_2 \leq 5 \\ x_2 \leq 2 \end{array}\right\},$$

$x_j \geq 0$ – бүтін, $(j = 1{,}2)$

17. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = x_1 + x_2 \to \max$$
$$\left.\begin{array}{l} 2x_1 + x_2 \le 4 \\ x_1 + 2x_2 \le 4 \end{array}\right\},$$
$$x_1 \ge 0, \quad x_2 \ge 0 - \text{бүтін}$$

18. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = x_1 + 2x_2 \to \max$$
$$\left.\begin{array}{l} x_1 + x_2 \le 4 \\ x_2 \le 2{,}8 \end{array}\right\},$$
$$x_j \ge 0 - \text{бүтін}, \ (\text{j} = 1{,}2)$$

19. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = x_1 + 2x_2 \to \max$$
$$\left.\begin{array}{l} 3x_1 + x_2 \le 7 \\ x_1 + 3x_2 \le 7 \end{array}\right\},$$
$$x_j \ge 0 - \text{бүтін}, \ (\text{j} = 1{,}2)$$

20. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = 110x_1 + 90x_2 \to \max$$
$$\left.\begin{array}{l} 3x_1 + 4x_2 \le 10 \\ 2x_1 + \ x_2 \le 8 \\ \qquad\ x_2 \le 5 \end{array}\right\},$$
$$x_j \ge 0 - \text{бүтін}, \ (\text{j} = 1{,}2)$$

21. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = x_1 + 4x_2 \to \max$$
$$\left.\begin{array}{l} 2x_1 + 4x_2 \le 17 \\ 10x_1 + 3x_2 \le 15 \end{array}\right\},$$
$$x_j \ge 0 - \text{бүтін}, \quad (\text{j} = 1,2)$$

22. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = x_1 + 4x_2 \to \max$$
$$\left.\begin{array}{l} 2x_1 + 4x_2 \le 7 \\ 10x_1 + 3x_2 \le 15 \end{array}\right\},$$
$$x_j \ge 0 - \text{бүтін}, \quad (\text{j} = 1,2)$$

23. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = x_1 + x_2 \to \max$$
$$\left.\begin{array}{l} 6x_1 + 5x_2 \le 20 \\ 2x_1 + 3x_2 \le 10 \end{array}\right\},$$
$$x_j \ge 0 - \text{бүтін}, \quad (\text{j} = 1,2)$$

24. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = x_1 + 2x_2 \to \max$$
$$\left.\begin{array}{l} 2x_1 + 9x_2 \le 36 \\ x_1 + x_2 \le 7 \end{array}\right\},$$
$$x_j \ge 0 - \text{бүтін}, \quad (\text{j} = 1,2)$$

25. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = 6x_1 + 4x_2 \to \min$$
$$\left.\begin{array}{l} 2x_1 + x_2 \ge 3 \\ x_1 - x_2 \ge 1 \end{array}\right\},$$
$$x_j \ge 0 - \text{бүтін}, \quad (\text{j} = 1,2)$$

26. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = 5x_1 + 7x_2 \to \min$$

$$\left.\begin{array}{l} -3x_1 + 14x_2 \le 78 \\ 5x_1 - 6x_2 \le 26 \\ x_1 + 4x_2 \ge 25 \end{array}\right\}$$

$$x_1,\ x_2 \ge 0$$

$x_1,\ x_2$ – бүтін

27. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = 2{,}5x_1 + 4x_2 + 3x_3 \to \max$$

$$\left.\begin{array}{l} 4{,}5x_1 + 3x_2 + 5x_3 \le 14 \\ 2x_1 + 6{,}3x_2 + x_3 \le 11 \end{array}\right\},$$

$$x_1, x_2, x_3 \ge 0,\ x_1, x_3 - \text{бүтін}$$

28. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = x_1 - x_2 \to \max$$

$$\left.\begin{array}{l} x_1 - 2x_2 + x_3 = 1 \\ x_1 + 3x_2 + x_4 = 3 \end{array}\right\},$$

$$x_j \ge 0 - \text{ бүтін};\ \left(j = \overline{1,4}\right)$$

29. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = x_4 - x_5 \to \max$$

$$\left.\begin{array}{l} x_1 + x_4 - 2x_5 = 1 \\ x_2 - 2x_4 + x_5 = 2 \\ x_3 + 3x_4 + x_5 = 3 \end{array}\right\},$$

$$x_j \ge 0 - \text{бүтін};\ \left(j = \overline{1,5}\right)$$

30. Гомори және Ленд-Дойг әдістерін қолданып келесі есепті шығарыңыз:

$$z = 2x_1 - 8x_2 \to \max$$

$$\left.\begin{array}{l} x_1 + 2x_2 \le 16 \\ 6x_1 + 2x_2 \le 40 \end{array}\right\}$$

$$x_1,\ x_2 \ge 0$$

$x_1,\ x_2$ – бүтін

6. ДИНАМИКАЛЫҚ БАҒДАРЛАМАЛАУ

6.1. Динамикалық бағдарламалау туралы жалпы мәліметтер және негізгі ұғымдар

Динамикалық бағдарламалау деген сөз тіркесін 1940-жылдары Р.Беллман шешімі өзінің «алдындағы» есепті шешкеннен кейін ғана алынатын есептің шешімін табу процесін сипаттау үшін қолданған. 1953 жылы ол бұл анықтаманы қазіргідей мағынасына дейін нақтылады. Бастапқыда бұл сала жүйелік талдау мен инжиниринг ретінде құрылған болатын. Беллманның динамикалық бағдарламалауға қосқан үлесі Беллман теңдеуінің аты арқылы мәңгі есте қалады, бұл теңдеу динамикалық бағдарламалау теориясының негізгі нәтижесі болып табылады, ол оңтайландырушы есепті рекурсивті формада қайта тұжырымдайды.

Динамикалық бағдарламалау – бұл математикалық бағдарламалау бөлімі, атап айтқанда, оңтайлы шешімдерді табуға мүмкіндік беретін амалдар жиынтығы, олар әрбір шешімнің нәтижелерін есептеуге және кейінгі шешімдер үшін оңтайлы стратегия құруға негізделген.

Осындай принцип арқылы құрылатын шешім қабылдау процестері көп қадамды процестер деп аталады. Оңтайландырушы есеп математикалық түрде бір-бірімен өзара тізбектей байланысқан осындай қатынастардың көмегімен құрылады: мысалы, бір жылға арналып алынған нәтиже келесі жылға (немесе керісінше, алдыңғы жылға) арналған теңдеуге енгізіледі және т.б. Осылайша, ЭЕМ көмегімен кез келген таңдалған уақыт мезгілі үшін есеп шешімінің нәтижелерін алып, ары қарай «жүріп отыруға» болады. Динамикалық бағдарламалау уақыт ағынымен байланысты есептерде ғана қолданылуы міндетті емес. Толық статикалық есепті шешу процесі де көп қадамды болуы мүмкін. Мысалы, ресурстарды үлестірудің кейбір есептері де осындай.

Динамикалық бағдарламалау есептері үшін ортақ нәрсе: ондағы айнымалылар бірге емес, бірінен кейін бірі тізбектей

қарастырылады. Мұндай есептеу сұлбасын құру кезінде, көп айнымалысы бар бір есептің орнына әрқайсысында аз айнымалысы (әдетте тіпті бір айнымалысы) бар көп есеп құрылады. Бұл есептеулердің көлемін айтарлықтай қысқартады. Алайда мұндай артықшылыққа тек екі шарт орындалғанда ғана қол жеткізуге болады:

1. оңтайлылық критерийі аддитивті болған кезде, яғни жалпы оңтайлы шешім әрбір қадамның оңтайлы шешімдерінің қосындысы болған кезде;

2. болашақ шешімдер жүйенің шешім қабылданып жатқан күйіне тәуелді болмаған кезде.

Бұл динамикалық теория негізінде жатқан Беллманның оңтайлылық принципінен шығады. Одан негізгі амал да шығады, ол – басымдық ережелерін табу, осының негізінде әрбір қадамда болашақ даму нұсқалары салыстырылады және келешегі жоқ нұсқалар алдын-ала жойылып отырады. Осы ережелер бірінен-бірі шығатын элементтер тізбегін анықтайтын формулаларға айналған кезде, олар шешуші ережелер деп аталады. Есептеулерді қысқарту арқылы ұтқанымызбен, олардың көлемі өте үлкен болып қала береді.

«Көлемінің үлкендігі» болмағанда, осы әдісті қолдануға болатын есептердің екі класын бөліп көрсетуге болады (негізінде, барынша жеңілдетілген түрде алынған осындай есептерде әдістің жалпы негіздерін тек қана көрсетіп, экономикалық-математикалық модельдерін талдай аламыз). Біріншісі – өндірілетін өнімнің уақыт ішіндегі қажеттілігін ескере отырып, экономикалық объекттің (кәсіпорынның, саланың және т.с.с.) қызметін жоспарлау есебі. Есептердің екінші класы – ресурстарды түрлі салалар арасында уақыт ішінде оңтайлы үлестіру. Мұнда, атап айтқанда, мынадай есепті жатқызуға болады: бірнеше жылда нанның ең көп мөлшерін алуға болатындай етіп, әрбір жылдың астық өнімін қорек пен дәнге қалай үлестіру керек?

Динамикалық бағдарламалау есепте кезең бойынша шешім қабылдау керек болған жағдайда айрықша тиімді қолданылады. Оны жабдықты жөндеу және ауыстыру есебінің қарапайым мысалында көрсетейік. Шина жасау зауытының құю

машинасында бір уақытта екі шина екі формада дайындалады. Егер бір форма істен шықса, машинаны бөлшектеу керек болады. Бұл жерде екінші форма келесі кезеңде істен шығып қалған жағдайда машинаны тағы да бөлшектеп жүрмеу үшін екінші форманы да ауыстыру тиімді болуы мүмкін. Және тағы бір жағдай – ешқайсысы істен шықпай тұрып екеуін де ауыстыру тиімді болуы мүмкін. Динамикалық бағдарламалау әдісінің көмегімен (шешімдер бұтағының көмегімен) барлық факторларды: форманы пайдалануды жалғастырудан болатын пайданы, машинаның тұрып қалуынан келетін шығындарды, ақаулы шиналардың бағасын және т.б. ескере отырып формаларды ауыстыру туралы мәселедегі ең жақсы стратегия анықталады.

Әдетте динамикалық бағдарламалау қарапайым іріктеу арқылы (уақыттық шектеулерге байланысты немесе тым үлкен жады көлемін қажет ететіндігіне байланысты) және сараң алгоритмдер (олар оңтайлы нәтиже бермейтіндіктен) арқылы шешілмейтін, ал есепті шешудің жеңілірек әдісі бар болса да оның табылуы анық емес болған оңтайландыру есептеріне қолданылады. Мұндай есептердің көп шешімі болуы мүмкін және жалпы жағдайда оны қандай да бір көрсеткіш анықтайды (оны есеп сапасы деп атайық), және осы көрсеткіш не минимум, не максимум болатындай (есептің шартына байланысты) оңтайлы шешімді таңдау керек.

Динамикалық бағдарламалау принципі бөліп ал да билей бер әдісіне ұқсайды: есеп ішкі есептерге бөлінеді, олар не айқын есептер болады, не болмаса өздерінің ішкі есептеріне келтіріледі және т.б. Бірақ кейбір айырмашылықтары да бар, мысалы, әдетте мұндай ішкі есептер салыстырмалы түрде аз болады, бұл әрбір есепті тек бір рет қана шешіп, нәтижесін (сапасын) қандай да бір массивке жазуға мүмкіндік береді, мұндай тәсіл memoization деп аталады.

Ішкі есептердің саны аз болады, бірақ олардың көпшілігін бірнеше рет шығаруға тура келеді, олар ішкі есептердің қайта жабылуы деп аталады және бұл динамикалық бағдарламалаудың сипатты белгісі болып табылады.

Қарастырылып отырған әдістің екінші сипатты белгісі ішкі есептердің оңтайлылығы болып табылады. Егер есептің оңтайлы шешімі оның ішкі есептерінің оңтайлы шешімдерін қамтыса, онда есеп осы қасиетке ие деп айтылады.

Динамикалық бағдарламалау принципі бойынша алгоритм құрудың жалпы сипаты төмендегідей:

– есептің шартын жақсылап түсіну;

– есептің оңтайлы шешімі өзіне кіретін барлық ішкі есептердің оңтайлы шешімдерін қамтитындай етіп, есепті екі не одан да көп ішкі есептерге бөлу;

– рекуррентік қатынас жазу (егер біз есепті ішкі есептерге бөлетін болсақ, демек, есептің сапасын оның ішкі есептері арқылы, немесе қарапайым түрде өрнектей аламыз);

– бүкіл есеп үшін параметрдің оңтайлы мәнін есептеу.

Есепті шешудің екі түрі бар:

1. Егер есеп рекурсивті түрде шығарылатын болса, қандай да бір процедура арқылы есеп ішкі есептерге бөлінеді. Бұл жерде алдымен есептің тривиал еместігі анықталады және оның бұрын шығарылмағандығы тексеріледі (бұрын шығарылған болса, тиісті массивтен дайын шешімді ала салу керек). Осы ішкі есептер шешіледі. Мұндай шешу түрі «жоғарыдан төмен қарай» деп аталады, яғни глобальды есепті аламыз да, сосын оған қажетті ішкі есептерді ғана шығарамыз.

2. Егер техникалық немесе басқа да себептерге байланысты рекурсия мүмкін болмаса, онда мынадай тәсіл қолданылады: алдымен қарапайым ішкі есептер шешіледі, сосын шешіліп қойған ішкі есептердің нәтижелерін қажет ететін ішкі есептер және т.б., осылайша жалпы есеп шешілгенше жалғасады. Мұндай әдіс «төменнен жоғары қарай» деп аталады. Көпшілік жағдайда ол жылдам болып келеді, бірақ түсініксіздеу және қарапайым болып табылады, дегенмен ерекше жағдайлары да бар.

Есепті шешудің қай нұсқасы қолданылса да, бастапқы есептің оңтайлы шешімін алу үшін соңғы қадамда әрбір ішкі есептің шешілу барысы туралы қосымша ақпарат қажет. Әрбір ішкі есептің алынған нәтижелерін пайдалан отырып, бастапқы

есептің шешімі анықталады. Бұл қадам «кері жүріс» деп аталады.

Динамикалық бағдарламалаудағы оңтайлы ішкі құрылым дегеніміз кіші көлемді ішкі есептердің оңтайлы шешімі бастапқы есепті шешу үшін қолданыла алады дегенді білдіреді. Мысалға, графтағы бір төбеден (s деп белгілейік) екінші төбеге (t деп белгілейік) дейінгі ең қысқа жол келесідей табылуы мүмкін: алдымен s-пен көршілес барлық төбелерден t-ға дейінгі ең қысқа жолды санаймыз, содан соң, s-тің көршілес төбелермен жалғанған қабырғаларының салмақтарын ескере отырып, t-ға дейінгі ең жақсы жолды таңдаймыз (қай төбе арқылы жүрген дұрыс). Жалпы жағдайда оңтайлы ішкі құрылымы бар есепті келесі үш қадамды орындай отырып шешуімізге болады:

– есепті кіші көлемді ішкі есептерге бөлу .

– дәл осындай үш қадамды алгоритмді орындай отырып, ішкі есептердің оңтайлы шешімін рекурсивті түрде табу.

– ішкі есептердің алынған шешімін бастапқы есептің шешімін құрастыру үшін қолдану.

Ішкі есептер оларды одан да кіші ішкі есептерге, есептің константты уақытта шешілетін тривиал жағдайына (жауабын бірден айтуға болатын жағдайға) жеткенше бөлу арқылы шығарылады. Мысалы, егер бізге n! табу керек болса, онда оның тривиал есебі 1! = 1 (немесе 0! = 1) болады.

Динамикалық бағдарламалаудағы қайта жабылатын ішкі есептер дегеніміз көлемі үлкенірек бірнеше (бір емес) есептерді шешу үшін қолданылатын ішкі есептерді білдіреді (яғни біз бір нәрсені бірнеше рет орындаймыз). Соның айқын бір мысалы Фибоначчи қатарын есептеу болып табылады, F3 = F2 + F1 және F4 = F3 + F2 – екі-ақ Фибоначчи санын есептеудің осындай тривиал жағдайының өзінде біз F2-ні екі рет санадық. Егер осыны жалғастырып, F5-ті санайтын болсақ, F2 тағы екі рет саналады, себебі F5-ті есептеу үшін қайтадан F3 пен F4 керек болады. Бұдан шығатыны: қарапайым рекурсивтік тәсіл өзі шығарған есептерді шешуге уақыт шығындайды.

Осындай жағдайды болдырмау үшін біз өзіміз есептеген ішкі есептердің шешімдерін сақтап отыратын боламыз, және ішкі есептің шешімі қайта қажет болған жағдайда біз оны жай

ғана жадыдан ала саламыз. Бұл тәсіл кэштеу деп аталады. Бұдан ары қарай да оңтайландыруға болады – мысалы, егер бізге ішкі есептің шешімі қажет болмайтындығына анық көзіміз жетіп тұрса, жадыны басқа қажеттіліктерге босату үшін оны жадыдан алып тастауға болады, немесе егер процессор жұмысы баяулап қалса және әлі есептелмеген кейбір ішкі есептердің шешімі кейінірек қажет болатынын білсек, біз оларды алдын-ала шешіп ала аламыз.

Жоғарыда айтылғандарды қорытындылай келе, динамикалық бағдарламалау есептің төмендегідей қасиеттерін қолданатындығын айтуға болады:

– қайта жабылатын ішкі есептер;

– оңтайлы ішкі құрылым;

– жиі кездесетін ішкі есептердің шешімдерін есте сақтау мүмкіндігі.

Динамикалық бағдарламалау әдетте есепті шешудің екі тәсілін ұстанады:

– төмендеуші динамикалық бағдарламалау: есеп көлемі кішірек ішкі есептерге бөлінеді, олар шешіліп, сосын бастапқы есепті шешу үшін біріктіріледі. Жиі кездесетін ішкі есептердің шешімдерін жадыда сақтау мүмкіндігі бар.

– жоғарылаушы динамикалық бағдарламалау: бастапқы есепті шешуге қажет болатын барлық ішкі есептер есептеледі де, содан соң бастапқы есептің шешімін құру үшін қолданылады. Бұл тәсіл төмендеуші бағдарламалаумен салыстырғанда жақсырақ болып келеді, себебі қажетті стек өлшемі мен функцияларды шақыру саны аз болады, бірақ бізге қандай ішкі есептердің қажет болатындығын алдын-ала анықтау кейде мүмкін болмайды.

Динамикалық бағдарламалау көмегімен шешілетін есептердің негізгі қасиеттерінің бірі аддитивтілік болып табылады. Аддиттивті емес есептер басқа әдістермен шешіледі. Мысалы, компания инвестицияларын оңтайландыру есептерінің көпшілігі аддитивті емес болып табылады және олар инвестиция жүргізу кезінде және инвестициясыз компания бағасын салыстыру арқылы шешіледі.

6.2. Динамикалық бағдарламалау әдісінің есептері

Динамикалық бағдарламалауда жиі қолданылатын есептер:

– Ең үлкен жалпы ішкі тізбек туралы есеп: екі тізбек берілген, ең ұзын жалпы ішкі тізбекті табу керек.

– Ең үлкен артушы ішкі тізбекті іздеу есебі: тізбек берілген, ең ұзын өспелі ішкі тізбекті табу керек.

– Түзету қашықтығы туралы есеп, (Левенштейн қашықтығы): екі жол берілген, бір жолды екінші жолға түрлендіретін символдардың өшірілуінің, ауыстырылуының және қосылуының минимал санын табу.

– Фибоначчи сандарын есептеу туралы есеп.

– Матрицаларды өзара көбейту реті туралы есеп: A_1, ..., A_n матрицалары берілген, оларды өзара көбейтуге арналған скалярлы операциялар санын минималдау керек.

– Траекторияны таңдау туралы есеп.

– Тізбектей шешім қабылдау есебі.

– Жұмыс күшін пайдалану туралы есеп.

– Ресурстарды үлестіру есебі.

– Қорларды басқару есебі.

– Ранец туралы есеп: «құны» және «салмағы» деген қасиеттері бар заттардың шексіз жиынынан шектеулі жиынтық салмақ жағдайында максимал жиынтық құнын алуға болатындай етіп заттарды іріктеп алу керек.

– Өлшенген бағытталған графтың барлық төбелері арасындағы ең қысқа қашықтықты табу есебі (Флойд-Уоршелл алгоритмі).

– Өлшенген графтың берілген екі төбелері арасындағы ең қысқа жолды табу есебі (Беллман-Форд алгоритмі).

Оқулықтың бұл тарауында капитал салымдарын үлестіру туралы есеп, ресурстарды үлестіру туралы есеп (қорларды басқару есебі), ұшақты жүктеу туралы есеп (ранец туралы есеп) қарастырылған.

Кейінгі параграфтарда динамикалық бағдарламалаудың жалпы есебінің математикалық қойылымы, оңтайлы басқаруды кезең бойынша құру принципі, математикалық модельді құру ережелері, ресурстарды үлестіру, қорларды басқару және ранец

туралы есептерді көрсете отырып динамикалық бағдарламалаудың есептеу сұлбасы ұсынылған.

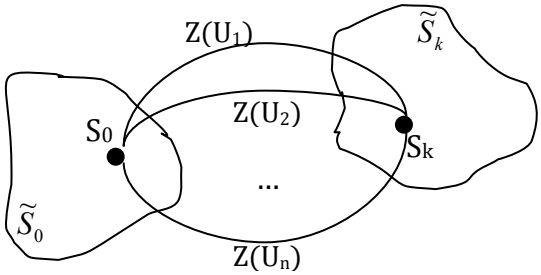
6.3. Динамикалық бағдарламалау есебінің жалпы қойылымы

Қандай да бір S жүйесі бастапқы күйде $S_0 \in \widetilde{S_0}$ тұр дейік. Уақыт өте келе оның күйі өзгереді де, жүйе ақырғы күйге $S_k \in \widetilde{S_k}$ ауысады. Жүйе күйінің өзгеру процесімен қандай да бір Z критерийі байланысты.

Тапсырма мынадай: U мүмкін басқарулар жиынының ішінен S жүйесін S_0 күйінен S_k күйіне көшіруге мүмкіндік беретіндей U^* басқаруын табу керек, мұнда Z(U) критерийі Z^* оңтайлы мәнін қабылдауы тиіс.

6.4. Оңтайлы басқаруды кезең бойынша құру принципі

Динамикалық бағдарламалау дегеніміз көп қадамды процесті кезең бойынша жоспарлау болып табылады, осы жоспарлаудың әрбір кезеңінде бүкіл процестің дамуы ескеріле отырып, тек бір қадам ғана оңтайланады.



17-сурет

Соңғы k қадамы үшін шешім қабылдау болашақ қадамға тәуелді емес, осы қадамда ең үлкен тиімділікті алуға мүмкіндік беретін басқару таңдалады. Осы қадамды жоспарлаған соң, оған

оның алдындағы (k-1)-ші қадамды қосуға болады, ал бұған (k-2)-қадам және т.б.

Ақырында S_0 бастапқы күйі табылады.

k-қадамды жоспарлау үшін жүйенің (k-1)-қадамдағы күйін білу керек. Егер жүйенің (k-1)-қадамдағы күйі белгісіз болса, онда берілген процестің сипатына қарап жүйенің берілген қадамдағы мүмкін күйлері туралы түрлі болжамдар жасалады: жүйенің n-шы қадамның соңындағы күйі $(n = \overline{1,(k-1)})$ тек алдыңғы S_{n-1} күйіне және берілген қадамдағы U_n басқаруына тәуелді:

$$S_n=F(S_{n-1}, U_n). \qquad (6.1)$$

Жүйенің мұндай қасиеті кейінгі әрекеттің болмауы деп, ал (6.1) теңдеуі күй теңдеуі деп аталды.

Әрбір S_{k-1} болжамы үшін соңғы k-шы қадамдағы оңтайлы басқару таңдалады $U_k^*(S_{k-1})$:

$$Z_k^*(U_k^*(S_{k-1}))=extr_{Uk \in Dk} f_k(U_k, S_{k-1}), \qquad (6.2)$$

мұндағы $f_k(U_k, S_{k-1})$ – k-шы қадамның тиімділік көрсеткіші, D_k – мүмкін басқарулар облысы.

Мұндай оңтайлы басқару «шартты оңтайлы басқару» деп аталады. (k-1)-қадамда да дәл осылай жасаймыз, бірақ шартты-оңтайлы басқаруды k-шы қадамда таңдалған шартты оңтайлы басқаруларды ескере отырып таңдау керек:

$$Z_n^*(U_n^*(S_{n-1}))=extr_{Un \in Dn}\{f_n(U_n, S_{n-1})+Z_{n+1}(U_{n+1}(S_n))\}, n=k-1, k-2, \ldots \quad (6.3)$$

мұндағы (6.3) теңдеуі – динамикалық бағдарламалаудың негізгі функционалдық теңдеуі немесе Беллман теңдеуі және ол берілген n-шы қадамнан k-шы қадамға дейінгі жиынтық тиімділікті сипаттайды.

Нәтижесінде $S_0 \in \widetilde{S_0}$ бастапқы күйге келеміз. Бірінші қадам үшін жүйенің мүмкін күйі туралы болжам жасамаймыз, себебі бұл S_0 күйі белгілі, тек U^* оңтайлы басқаруды табамыз.

S_0-ден S_k-ға жүре отырып бүкіл процес үшін ізделінген оңтайлы басқаруды табамыз.

6.5. Динамикалық бағдарламалаудың есептеу сұлбасы

1^0. (6.3) теңдеуінің шешімі (6.2) теңдеуінен бастап тізбектей орындалады – шартты оңтайландыру кезеңі.

2^0. k жеке есептерді шартты extr-ға тізбектей шешу нәтижесінде функциялардың екі тізбегі анықталады:

$$Z_n^*(U_n^*(S_{n-1})), \qquad U_n^*(S_{n-1}).$$

Дискретті есептерде бұл функциялар кесте түрінде алынады, ал үздіксіз есептерде аналитикалық түрде алынады.

3^0. Шартты оңтайландыру кезеңін орындаған соң (I кезең) II кезеңге – шартсыз оңтайландыру кезеңіне өтеміз:

а) егер S_0 берілген болса, мынаны табамыз: $Z^*=Z_1^*(U_1^*(S_0))$, (6.4) сосын тізбек бойынша ізделінді шартсыз оңтайлы басқаруды табамыз:

$$S_0 \to U_1^* \to S_1 \to U_2^* \to \ldots \to U_k^*. \qquad (6.5)$$

б) егер S_0 белгісіз болса, онда S_0^* келесі формуламен табылады:

$$Z^*=\text{extr}_{S_0 \in \widetilde{S_0}}\{Z_1^*(U_1^*(S_0))\}, \qquad (6.6)$$

сосын (6.5) теңдеуі бойынша шартсыз оңтайлы басқаруды табамыз.

6.6. Динамикалық бағдарламалаудың математикалық моделін құру ережелері

1. Қадамдарға бөлу тәсілдері таңдалады.
2. Күй параметрлері $S_n=(S_{n1},\ldots,S_{np})$ және басқару айнымалылары $U_n=(U_{n1},\ldots,U_{nr})$ енгізіледі.
3. (6.1) күй теңдеуі жазылады.

4. n-қадамда тиімділік көрсеткіші және мақсатты функцияның тиімділігінің жиынтық көрсеткіші енгізіледі:

$$Z = \sum_{n=1}^{k} f_n(U_n, S_{n-1}). \qquad (6.7)$$

5. Шартты extr $Z_n^*(U_n^*(S_{n-1}))$ қарастырылу үшін енгізіледі, яғни n-ші қадамнан бастап процестің соңына дейінгі тиімділік көрсеткіші, осы қадамдағы шартты- оңтайлы басқаруға қарай $U_n^*(S_{n-1})$.

6. Есептің шектеуінен берілген қадам үшін D_n анықталады – ол n-ші қадамның қосымша басқаруларының жиыны .

7. Беллманның (6.2) және (6.3) функционалдық теңдеулері жазылады.

13-мысал. Қаражатты үлестіру (үздіксіз есеп).

2 кәсіпорынның 4 жыл бойғы қызметі жоспарлануда. 1-ші кәсіпорынға салынған $Z_0 = \xi_0 = 1000$ бастапқы қаражат жылдың соңына қарай $f_1(x) = 0.4x$ табыс алып келеді және $\varphi_1(x) = 0.5x$ көлемде қайтарылады, 2-ші кәсіпорын үшін $f_2(y) = 0.3y$, $\varphi_2(y) = 0.8y$.

Бір жыл өткен соң қалған қаражаттың барлығы екі кәсіпорын арасында қайта үлестіріледі. Жаңа қаражат түспейді және алынған табыс өндіріске салынбайды. Қолда бар қаражатты үлестірудің оңтайлы тәсілін анықтау керек.

Шешуі:

4 жылды 4 кезеңге бөліп тастайық, әрбір жылға сәйкесінші бір кезеңді қоямыз.

k=4, n=4, 3, 2, 1.

x_k – 1-ші кәсіпорынға k-шы жылы бөлінетін қаражат.

$y_k = \xi_{k-1} - x_k$ – екінші кәсіпорын (шешуінде бір айнымалы болу үшін оны x_k арқылы өрнектейік).

Екі кәсіпорыннан алынған жалпы шығын мынаған тең:

$$z_k = 0.4k_x + 0.3(\xi_{k-1} - x_k) = 0.1x_k + 0.3\xi_{k-1}$$

Күй теңдеуі k-шы жылдан кейінгі ξ_k қаражат қалдығын өрнектейді:

$$\xi_k = 0.5x_k + 0.8(\xi_{k-1} - x_k) = 0.8\xi_{k-1} - 0.3x_k$$

Негізгі функционалдық теңдеу:

$$z_4^*(\xi_3) = \max_{0 \le x_4 \le \xi_3}\{0.1x_4 + 0.3\xi_3\}$$

k-шы жылдан бастап қарастырылып отырған мерзімнің соңына дейінгі екі кәсіпорын арасында n-k+1 жылда ξ_{k-1} қаражатты үлестіруден түскен шартты оңтайлы табыс:

$$z_k^*(\xi_{k-1}) = \max_{0 \le x_k \le \xi_{k-1}}\{0.1x_k + 0.3\xi_{k-1}\} + z_{k+1}^*\{0.8\xi_{k-1} + 0.3x_k\}.$$

Шартты оңтайландыру кезеңі:

1) $z_4^*(\xi_3) = \max_{0 \le x_4 \le \xi_3}\{0.1x_4 + 0.3\xi_3\} = \max\{0.3\xi_3; 0.4\xi_3\} = 0.4\xi_3,$

функция $x_4 = \xi_3$ интервалының соңында max болатындықтан, $x_4^*(\xi_3) = \xi_3$ болады.

2) $$z_3^*(\xi_2) = \max_{0 \le x_3 \le \xi_2}\{0.1x_3 + 0.3\xi_2 + 0.4(0.8\xi_2 - 0.3x_3)\} = \max_{0 \le x_3 \le \xi_2}\{-0.02x_3 + 0.62\xi_2\} = 0.62\xi_2,\ x_3^*(\xi_2) = 0.$$

3) $$z_2^*(\xi_1) = \max_{0 \le x_2 \le \xi_1}\{0.1x_2 + 0.3\xi_1 + 0.62(0.8\xi_1 - 0.3x_2)\} = \max_{0 \le x_2 \le \xi_1}\{-0.086x_2 + 0.796\xi_1\} = 0.796\xi_1,\ x_2^*(\xi_1) = 0.$$

4) $$z_1^*(\xi_0) = \max_{0 \le x_1 \le \xi_0}\{0.1x_1 + 0.3\xi_0 + 0.796(0.8\xi_0 - 0.3x_1)\} = \max_{0 \le x_1 \le \xi_0}\{-0.1388x_1 + 0.9368\xi_0\} = 0.9368\xi_0,\ x_1^*(\xi_0) = 0.$$

Шартсыз оңтайландыру кезеңі:

$\xi_0^* = 10000,\ z_1^* = 9368,\ x_1^* = 0,$ қалдығы

$\xi_1 = 0.8\xi_0 - 0.3x_1 = 0.8 * 10000 - 0.3 * 0 = 8000$

$z_2^* = 0.796 * 8000 = 6368,\ x_2^* = 0,$ қалдығы

$\xi_2 = 0.8 * 8000 - 0.3 * 0 = 6400$

$z_3^* = 0.62 * 6400 = 3968,\ x_3^* = 0,$ қалдығы

$\xi_3 = 0.8 * 6400 = 5120$

$z_4^* = 0.4 * 5120 = 2048,\ x_4^* = 5120$

Оңтайлы шешім 90-кестеде келтірілген:

90-кесте

Кәсіпорын	Жылдар				
	I	II	III	IV	
I	0	0	0	5120	x_k
II	10000	8000	6400	0	$y_k = \xi_{k-1} - x_k$

14-мысал. Қаражатты үлестіру (дискретті есеп).

4 кәсіпорын бар, оларға 100 мың теңге үлестіріліп берілуі тиіс. Бөлінген x сомасына байланысты кәсіпорындардағы өнім шығарылуының өсу мәндері $f_i(x)$ 91-кестеде келтірілген. Өнім шығарылуының жалпы өсуін максималдайтындай қаражатты үлестіру жоспарын құру керек.

91-кесте

x	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$
20	10	12	11	16
40	31	26	36	37
60	42	36	45	46
80	62	54	60	63
100	76	78	77	80

Ш е ш у і .

n = 1 болсын (бірінші қадам). Барлық қаражат бірінші кәсіпорынға бөлінеді деп алайық, онда бастапқы с сомасына байланысты 91-кестені ескере отырып $D_1(x)$ мәндерін аламыз (92-кесте).

92-кесте

$x_1^*(c)$	20	40	60	80	100
$D_1(c)$	10	31	42	62	76

n = 2 (екінші қадам). Енді барлық қаражат екі кәсіпорынға бөлінеді деп алайық, онда $D_2(c) = \max\{f_2(x) + D_1(c-x)\}$ формуласына сәйкес келесі 93-кестені аламыз.

93-кесте

c	0	20	40	60	80	100	$D_2(c)$	$x_2^*(c)$
20	0+10	12+0					12	20
40	0+31	12+10	26+0				31	0
60	0+42	12+31	26+10	36+0			43	20
80	0+62	12+42	26+31	36+10	54+0		62	0
100	0+76	12+62	26+42	36+31	54+10	78+0	78	100

$D_3(x)$ мәндерінің есептелуі 94-кестеде келтірілген, мұнда барлық қаражат үш кәсіпорын арасында үлестіріледі.

94-кесте

c	0	20	40	60	80	100	$D_3(c)$	$x_3^*(c)$
20	0+12	11+0					12	0
40	0+31	11+12	36+0				36	40
60	0+43	11+31	36+12	45+0			48	40
80	0+62	11+43	36+31	45+12	60+0		67	40
100	0+78	11+62	36+43	45+31	60+12	77+0	79	40

Төртінші қадамда қаражат 4 кәсіпорын арасында үлестіріледі (95-кесте):

95-кесте

c	0	20	40	60	80	100	$D_4(c)$	$x_4^*(c)$
100	0+79	16+67	37+48	46+36	63+12	80+0	85	40

Барлық нәтижелер 96-кестеде келтірілген.

96-кесте

c	$x_1^*(c)$	$D_1(c)$	$x_2^*(c)$	$D_2(c)$	$x_3^*(c)$	$D_3(c)$	$x_4^*(c)$	$D_4(c)$
0	0	0	0	0	0	0		
20	20	10	20	12	0	12		
40	40	31	0	31	40	36		
60	60	42	20	43	40	48		
80	80	62	0	62	40	67		
100	100	76	100	78	40	79	40	85

Кері жүріс:
$x_4 = 40$, $\xi_1 = 100 - 40 = 60$.
$x_3 = 40$, $\xi_2 = 60 - 40 = 20$.
$x_2 = 20$, $\xi_3 = 20 - 20 = 0$.
$x_1 = 0$, $\xi_4 = 0$.

Осылайша, $X^* = (0, 20, 40, 40)$, $z_{max} = 85$ мың теңге.
(Тексеру: 0+12+36+37=85).

6.7. Бір өлшемді ранец туралы есеп

Жүк көтерімділігі d-ға тең ранец бар, оны салмағы мен құны жағынан n түрлі j = 1, 2, …, n жүкпен толтыруға болады. j-ші жүк бірлігінің салмағы a_j, құны c_j. Ранецтің қолданылуы максимал болатындай етіп, жүктер жиынын таңдау керек.

Осылайша тұжырымдалған есептің математикалық моделі келесідей:

максималдау керек

$$F(x) = \sum_{j=1}^{n} c_j x_j \qquad (6.8)$$

Ол келесі шарттармен анықталған G облысында болады:

$$\sum_{j=1}^{n} a_j x_j \leq d, \qquad (6.9)$$

$$x_j \geq 0, \text{ бүтін, } j = 1, 2, \ldots, n. \qquad (6.10)$$

Бұл есептің шешімі динамикалық бағдарламалау әдісінің көмегімен табылуы мүмкін. Бұл алгоритмдегі нұсқаларды тізбектей талдау әдісінің негізгі идеясы келесідей:

Шешімді іздеу облысы бірінің ішіне бірі салынған облыстардың тізбегі ретінде қарастырылады, бұл тізбек нөл шамасының облысынан басталып, берілген облыспен аяқталады. n айнымалысы бар функцияны максималдау процесі бір айнымалысы бар функцияны максималдаудың n-қадамдық процесіне ауыстырылады.

Жүк көтерімділігі a ($0 \le a \le d$) болатын ранецті толтырудың k-шы қадамында алынатын максимал тиімділікті f_k (a) деп белгілейік. Есепті шешудің әрбір қадамында бір айнымалысы бар функция максималданатын болғандықтан, қадам нөмірін қарастырылған айнымалының нөмірімен белгілеуге болады. Шынымен де,

$$F(x) = \sum_{j=1}^{n} c_j x_j = c_n x_n + \sum_{j=1}^{n-1} c_j x_j ,$$

Сонда

$$\max F(x) = \max \sum_{j=1}^{n} c_j x_j = \max (c_n x_n + \sum_{j=1}^{n-1} c_j x_j).$$

$x_1, x_2, \ldots, x_n$ (6.8-6.10) облысынан таңдалады.

а) $x_n=0$ деп алайық, онда x_n тұрақты болған жағдайда былай жазуға болады:

$$\max F(x) = \max c_j x_j, \text{ мұндағы } x_j \in [0, d].$$

б) $x_n=1$ болғанда алатынымыз:

$$\max F(x) = c_n + \max \sum_{j=1}^{n-1} c_j x_j, \text{ мұндағы } x_j \in [0, d - a_1].$$

Екі жағдайда да $x_j \ge 0$, бүтін, $j = 1, 2, \ldots, n - 1$.

x_n-ң мәнін өзгерте отырып, $x_1, x_2, \ldots, x_{n-1}$ тиісті анықталу облысындағы x_n-ң мәні берілген жағдайдағы $(n - 1)$ айнымалысы бар функцияны максималдау процесінің нітижесі ретінде F(x)-ң әр түрлі мәндерін аламыз. Бірақ, жүк көтерімділігі d-ға тең ранец үшін n айнымалысы бар функцияға арналған максимал тиімділікті біз $f_n(d)$ деп белгілегендіктен, былай жазуға болады:

$$\max \sum_{j=1}^{n-1} c_j x_j = f_{n-1} (d - a_n x_n).$$

$x_j \geq 0$ болғандықтан, $0 \leq x_n \leq [\frac{d}{an}]$ үшін $0 \leq \sum_{j=1}^{n-1} a_j x_j \leq d-a_n x_n$ болады. Сонымен, былай жазуға болады:

$$F_n(d) = \max 0 \leq x_n \leq [\frac{d}{an}] [c_n x_n + f_{n-1} (d-a_n x_n)]. \qquad (6.11)$$

Егер f_0 (a) = 0, $0 \leq a \leq d$ деп алатын болсақ, онда n = 1, 2, … үшін (6.11) теңдеуі дұрыс келетіні анық.

(6.11) теңдеуі негізгі рекуррентік қатынас деп аталады. Ол n айнымалысы бар функцияны максималдау процесін бір айнымалысы бар функцияны максималдаудың тізбекті процедурасы ретінде қарастыруға мүмкіндік береді.

Есептеу алгоритмін былай жазайық:

1. k деп біз қарастырып қойған тәсілдердің санын белгілейік (бастапқыда k=1).

2. Ресурстардың өзгеру интервалын ұзындықтары Δ болатындай етіп m дискретті интервалдарға бөлейік, олардың әрқайсысына $f_k(l\Delta)$, (l=0, 1, …, m) функциясын есте сақтауға арнаған ұяшықтар енгіземіз.

3. β жұмыстық ұяшығына түскен табыстың ең үлкенін жазамыз (бастапқыда ол 0-ге тең).

4. $y_k(x)$ деп x ресурсы бар k-ші қадамдағы оңтайлы шешімді белгілейік (кез келген k үшін бастапқыда x = 0).

5. $y_k(x)$, $f_k(x-y_k a_k)$, $c_k y_k$ мәндерін пайдалан отырып, $c_k y_k + f_k(x-a_k y_k)$ есептейміз де, α ұяшығына жазамыз.

6. α-ны β ұяшығындағы санмен салыстырамыз. Егер β ұяшығындағы сан үлкен болса, онда 8-пунктке көшеміз. Егер α > β болса, онда 7-пунктке көшеміз.

7. $\beta := \alpha$, $\gamma := y_k(x)$.

8. $y_k(x)$-ті δ шамасына өзгертеміз, $y_k(x) = y_k(x) + \delta$.

9. Салыстырамыз: $y_k(x) > \frac{x}{ak}$? Егер «иә» болса, онда $y_k(x)$-ті өзгертуге болмайды, себебі $0 \leq y_k(x) \leq \frac{x}{ak}$, сосын 10-пунктке көшеміз. Егер $y_k(x) < \frac{x}{ak}$ болса, онда 5-пунктке көшеміз.

10. x бастапқы ресурстар үшін $y_k(x)$-ң барлық мүмкін мәндерін бітіріп тастағандықтан, алынған $f_k(x)$ және $y_k(x)$ ішінен осы табысқа алып келетін ең үлкенін тиісті есте сақтау ұяшықтарында есте сақтайық. Бұл – қарастырылған ресурстар аясындағы оңтайлы шешім.

11. x ресурсының бастапқы санын Δ шамасына арттырайық. Жаңа есеп аламыз, оның да жаңа ресурстардан бастап пайдалану тәсілдерінің саны k-ға тең.

12. Егер $x + \Delta \le d$ болса, онда 5-пунктке көшеміз. Егер $x + \Delta > d$ болса, онда ресурстарды пайдаланудың k тәсілдері үшін (ранецті толтырудың k заттары үшін) есепті шешу процесі аяқталды.

13. $f_{k-1}(x)$ массивінің орнына $f_k(x)$ массивін жібереміз.

14. k:=k+1 (k+1 зат үшін есептің шешімі).

15. $k \le n$, егер «ия» болса, онда x:=0 және 5-пунктке өту. Егер k>n болса, онда санақ аяқталды.

Жүргізілген есептеулер нәтижесінде біз n кесте аламыз, олардың әрқайсысы жалпы табысты анықтайды және ресурстарды қолдану тәсілдерінің саны тұрақты болғандағы осы табысқа жетудің оңтайлы тәсілін анықтайды.

Есепті шешудің екінші фазасы – құрылған оңтайлы шешімдер кестесінен таңдау жасау. Ол келесі сұлба бойынша жүзеге асырылады:

1. Ресурсты пайдаланудың n тәсілін және бастапқы ресурстардың x_0 санын қамтитын есепті қарастырайық. k ұяшығына тәсілдер санын, ал k_1 ұяшығына ресурстар санын сақтайық.

2. k кестеден k және k_1 ұяшықтарында көрсетілген шамалар үшін табысқа жетудің оңтайлы тәсілін анықтайық, ол $y_k(x)$ шамасы болсын.

3. $y_k(x)$-ті баспаға шығару. Баспаға шығарылған $y_k(x)$-ті α-ға сақтайық.

4. $k_1=k_1 - a_k y_k(x)$, k:=k – 1.

5. Талдау: Егер $k \neq 0$ болса, онда k_1-ң нөлге теңдігін тексерген жөн.

6. $k_1=$ $\begin{cases} \neq 0{,}2\text{-ші бөлімге көшу,} \\ = 0\text{, қалған тәсілдердің барлығы нөлдік тәсілдер.} \end{cases}$

Егер k = 0 болса, онда баспаға шығару процесі аяқталды.

15-мысал. Ұшақты жүктеу туралы есеп.

Ұшақ әр түрлі n затпен толтырылады. і типті (i=1,…,n) әрбір заттың салмағы мен құны 97-кестеде келтірілген. Ұшақтың максимал жүк көтерімділігі W=83. Салмағы ұшақтың максимал жүк көтерімділігінен аспайтын жүктің максимал құнын анықтау керек.

97-кесте

p_i	c_i
24	96
22	85
16	50
10	20

Шешуі. Енгізілетін айнымалылар: x_i - і-ші типті заттар саны. Есептің математикалық моделі:

$$z = \sum_{i=1}^{n} c_i x_i \rightarrow \max$$

Шектеулері:

$$\sum_{i=1}^{n} p_i x_i \leq W$$

x1, x2, x3 , x4 айнымалыларының мәндерін табу керек.

$f_n(W)$ функциясын табу үшін $f_{n-1}(W-p_{nxn})$-ті білу керек болғандықтан, есепті шешу кезінде f(W) функциясының $0 \leq W \leq 83$ мәндеріндегі мәндері қажет болады.

1-кезең. Ұшақ тек 1-ші типті заттармен толтырылсын дейік. Оларды [W/p1] мөлшерде тиеуге болады, сондықтан $x_1=0,1,2,3$.

Жүк салмағы p1=24 бірлік болғандықтан, онда $0 \leq W \leq 23$ болған кезде ұшаққа 1-ші типті заттың бір де біреуін тиеуге болмайды.

$24 \leq W \leq 47$ болғанда, бір затты тиеуге болады.

$48 \leq W \leq 71$ болғанда, екі затты тиеуге болады.

$72 \leq W \leq 83$ болғанда, үш затты тиеуге болады.

W, $f_1(W)$, және x_1 табылған мәндерін 98-кесте түрінде жазайық.

98-кесте

W	$f_1(W)$	X_1
0-23	0	0
24-47	96	1
48-71	192	2
72-83	288	3

X_1=0,1,2,3

2-кезең. Ұшақ 1-ші және 2-ші типті заттармен толтырылсын дейік. p_2=22 бірлік болғандықтан, $[W/p_2]=[83/22]$. x_2=0,1,2,3. Жүк көтерімділік шамасын p1 мен p_2-ні ескере отырып интервалдарға бөлеміз.

0≤W≤23 болғанда ұшаққа 1-ші типті затты да, 2-ші типті затты да тиеуге болмайды. Қалған интервалдардағы заттардың саны кестенің көмегімен және функционалды теңдеудің көмегімен табылады:

$$f_2(W) = \max_{0 \le x \le [W/p_2]}\{c_2x_2 = f_1(W - p_2x_2)\}$$

99-кесте

W	$f_2(W)$	X_2	X_1
0-21	0	0	0
21-23	85	1	0
24-43	96	0	1
44-45	170	2	0
46-47	181	1	1
48-65	192	0	2
66-67	255	3	0
68-69	266	2	1
70-71	277	1	2
72-83	288	0	3

46≤W≤47,

$$f_2(44) = \max_{0 \le x_2 \le [46/22]}\{85x_2 + f_2(44 - 22x_2)\} = \max\{85 \cdot 0 + f_1(44);\ 85 \cdot 1 + f_1(44 - 22);\ 85 \cdot 2 + f_1(0)\} = \max\{96; 181; 170\} = 181$$

x_2=2.

68≤W≤69,

$$f_2(68) = \max_{0 \le x_2 \le [68/22]}\{85x_2 + f_1(68 - 22x_2)\} = \max\{85 \cdot 0 + f_1(68); 85 \cdot 1 + f_1(46); 85 \cdot 2 + f_1(24); 85 \cdot 3 + f_1(2)\} = \max\{192; 181; 266; 255\} = 266$$

x_2=2.
70≤W≤71,

$$f_2(70) = \max_{0 \le x_2 \le [70/22]}\{85x_2 + f_1(70 - 22x_2)\} = \max\{85 \cdot 0 + f_1(70); 85 \cdot 1 + f_1(48); 85 \cdot 2 + f_1(22); 85 \cdot 3 + f_1(4)\} = \max\{192; 277; 170; 255\} = 277$$

x2=1.

3-кезең. Алдыңғы кезеңдегідей ұшақтың жүктелуі қарастырылады. Бірақ бұл кезеңде 1-ші, 2-ші және 3-ші типті заттар таңдалады.

Шешімі кестеде келтірілген:

100-кесте

W	f_3(W)	x3	x2	x1
0 - 15	0	0	0	0
16 - 21	50	3	0	0
22 - 23	85	0	1	0
24 - 31	90	0	0	1
32 - 37	100	2	0	0
38 - 39	135	1	1	0
40 - 43	146	1	0	1
44 - 45	170	0	2	0
46 - 47	181	0	1	1
48 - 61	192	0	0	2
62 - 63	231	1	2	0
64 - 65	242	1	0	2
66 - 67	255	0	3	0
68 - 69	266	0	2	1
70 - 71	277	0	1	2
72 - 83	288	0	0	3

$$f_3(W) = \max_{0 \le x_2 \le [W/p_3]}\{c_3x_3 + f_2(W - p_3x_3)\}$$

$f_3(62) = \max_{0 \le x_2 \le [62/16]}\{50x_3 + f_2(62 - 16x_3)\} = \max\{50 \cdot 0 + f_2(62); 50 \cdot 1 + f_2(46); 50 \cdot 2 + f_2(30); 50 \cdot 3 + f_2(14)\} = \max\{192; 231; 196; 150\} = 231$

x2=2, x3=1, x1=0.

4-кезең. Алдыңғы кезеңдегідей ұшақтың жүктелуі қарастырылады. Бірақ бұл кезеңде 1-ші, 2-ші, 3-ші және 4-ші типті заттар таңдалады.

Шешімі кестеде келтірілген:

101-кесте

W			f3(W)	x4	x3	x2	x1
0	-	9	0	0	0	0	0
10	-	15	20	1	0	0	0
16	-	21	50	0	1	0	0
22	-	23	85	0	0	1	0
24	-	31	96	0	0	0	1
32	-	33	105	1	0	1	0
34	-	37	116	1	0	0	1
38	-	39	135	0	1	1	0
40	-	43	146	0	1	0	1
44	-	45	170	0	0	2	0
46	-	47	181	0	0	1	1
48	-	55	192	0	0	0	2
56	-	57	201	1	0	1	1
58	-	59	212	1	0	0	2
60	-	61	220	0	1	2	0
62	-	63	231	0	1	1	1
64	-	65	242	0	1	0	2
66	-	67	255	0	0	3	0
68	-	69	266	0	0	2	1
70	-	71	277	0	0	1	2
72	-	79	288	0	0	0	3
80	-	81	297	1	0	1	2

$f_4(64) = \max_{0 \le x_2 \le [64/10]}\{20x_4 + f_3(64 - 10x_4)\} = \max\{20 \cdot 0 + f_3(64); 20 \cdot 1 + f_3(54); 20 \cdot 2 + f_3(44); 20 \cdot 3 + f_3(34); 20 \cdot 4 + f_3(24); 20 \cdot 5 + f_3(14); 20 \cdot 6 + f_3(4)\} = \max\{242; 212; 210; 160; 170; 100; 120\} = 242$

x4=0, x3=1, x2=0, x1=2.

Ұшақтың жүгі үшін ең оңтайлы нұсқа: үшінші заттың бір бірлігі және бірінші заттың екі бірлігі.

Сұрақтар мен тапсырмалар

1. Динамикалық бағдарламалау есебінің жалпы қойылымын тұжырымдаңыз.
2. Динамикалық бағдарламалаудың есептеу сұлбасын келтіріңіз.
3. Динамикалық бағдарламалаудың негізгі функционалдық теңдеуін келтіріңіз.
4. Қандай есептер динамикалық бағдарламалау әдістерімен шығарылады?
5. Қандай басқару шартты оңтайлы басқару деп аталады?
6. Кейінгі әрекеттің болмауы шартын тұжырымдаңыз.
7. Оңтайлы басқару стратегиясы деп нені айтамыз?
8. Қандай шарт аддитивтілік шарты деп аталады?
9. Динамикалық бағдарламалау есептері неліктен көп кезеңді болып келеді?
10. Динамикалық бағдарламалау моделін құру кезінде қандай ережелер қолданылады?
11. 4 кәсіпорын бар, оларға 100 мың теңге үлестіріліп берілуі тиіс. Бөлінген x сомасына байланысты кәсіпорындардағы өнім шығарылуының өсу мәндері $f_i(x)$ 1-кестеде келтірілген. Өнім шығарылуының жалпы өсуін максималдайтындай қаражатты үлестіру жоспарын құру керек.

x	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$
50	17	25	20	30
100	38	40	35	45
150	50	48	52	50
200	55	56	60	55
250	60	62	68	68

12. 2 кәсіпорынның 4 жыл бойғы қызметі жоспарлануда. 1-ші кәсіпорынға салынған қаражат жылдың соңына қарай $f_1(x) = 0,6*x$ табыс алып келеді және $\varphi_1(x) = 0,5*x$ көлемде қайтарылады, 2-ші кәсіпорын үшін $f_2(x)=0,3*x$, $\varphi_2(x) = 0,7*x$.

Бір жыл өткен соң қалған қаражаттың барлығы 2 кәсіпорын арасында қайта үлестіріледі. Жаңа қаражат түспейді және алынған табыс өндіріске салынбайды. Табыстың максимумдық шарты ескеріле отырып, $z_0=\xi_0=300$ мөлшердегі қолда бар қаражатты үлестірудің оңтайлы тәсілін анықтау керек.

13. Келесі есепті динамикалық бағдарламалау әдісімен шығару керек:

x	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$
20	9	11	16	13
40	18	19	32	27
60	24	30	40	44
80	38	44	57	69
100	50	59	70	73

14. Келесі есепті динамикалық бағдарламалау әдісімен шығару керек:

x	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$
15	9	8	12	7
30	18	19	25	15
45	29	30	51	52
60	41	47	58	59
75	60	58	69	60

15. Келесі есепті динамикалық бағдарламалау әдісімен шығару керек:

x	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$
20	9	11	13	12
40	29	19	27	30
60	37	28	37	42
80	44	46	48	50
100	60	58	69	60

16. Келесі есептерді динамикалық бағдарламалау әдісімен шығару керек:

Нұсқа	$f_1(x)$	$f_2(y)$	$\varphi_1(x)$	$\varphi_2(y)$	ξ_0	N
1.	$1 - e^{-x}$	$1 - e^{-2y}$	0,75x	0,3y	2	5
2.	0,5x	0,7y	0,5x	0,6y	10	5
3.	$0,6(x-1)^2$	$0,5y^2$	0,9x	0,1y	30	5
4.	$1 - e^{-2x}$	$1 - e^{-3y}$	0,7x	0,2y	4	5
5.	x^2	$2y^2$	0,6x	0,3y	10	4
6.	$3x^2$	$2y^2$	0,45x	0,8y	20	4
7.	x^2	$3y^2$	0,5x	0,2y	30	4
8.	$4x^2$	$2y^2$	0,3x	0,7y	10	4
9.	$2x^2$	$4y^2$	0,7x	0,5y	30	4
10.	$2x^2$	$3y^2$	0,8x	0,4y	20	4

7. ТОРАПТЫҚ ЖОСПАРЛАУ ЖӘНЕ БАСҚАРУ

7.1. Жалпы мәліметтер мен ұғымдар

Тораптық жоспарлау және басқару (ТЖБ) – бұл тораптық графиктерді қолдану арқылы ірі халық шаруашылық кешендерін, ғылыми зерттеулерді, жаңа бұйым түрлерін өндірудің конструкторлық және технологиялық даярлығын, құрылыс пен қайта құруды жоспарлау және басқару жүйесі. ТЖБ жүйесі жоспарланатын жұмыстар мен алынған нәтижелердің өзара байланысын орнатуға, жоспарды нақтырақ есептеуге, сондай-ақ оны дер кезінде түзетіп отыруға мүмкіндік береді.

ТЖБ-ң бастапқы идеялары 50-жылдардың соңында АҚШ-та құрылған болатын және тораптық талдаудың екі жүйесі түрінде жүзеге асырылған – PERT (Program Evaluation and Review Technique – бағдарламаларды және тексеру тәсілдерін бағалау) және CPM (Critical Path Method – критикалық жол әдісі). Кеңес Одағында тораптық жоспарлау жұмыстары 60-жылдарда басталды. Ол кезде тораптық жоспарлау әдістері құрылыста және ғылыми зерттемелерде қолданылды. Кейінірек тораптық әдістер халық шаруашылығының басқа да салаларында кеңінен қолданыла бастады.

Тораптық жоспарлау әдісін қолдану арқылы кәсіпорын дамуының стратегиялық жоспарының оңтайлы нұсқасын жасауға болады, ол оны жүзеге асыру барысында жұмыстар кешенін жедел басқаруға негіз болып табылады. Бұл жүйедегі негізгі жоспарлы құжат тораптық график немесе жай ғана торап болып табылады, ол стратегиялық жоспарлаудың ақырғы мақсатына жетуге қажетті барлық логикалық байланыстар мен орындалатын жұмыстардың нәтижелері көрсетілетін ақпараттық-динамикалық модель болып табылады. Тораптық графикте барлық қызмет түрлерін берілген немесе жоспарланған мерзімнен кешіктірмей аяқталуын қамтамасыз ету үшін қандай жұмыстар қандай ретпен және қанша уақытта орындалуы керектігі қажетті деңгейде талданып көрсетіледі. Тораптық

модельдер жоспарлаудың бастапқы кезеңінде құрылады. Алдымен жоспарланатын процес бөлек жұмыстарға бөлінеді, жұмыстар мен оқиғалар тізімі құрылады, олардың логикалық байланыстары мен олардың орындалу реті ойластырылады, жұмыстар жауапты орындаушыларға бекітіледі. Осылардың көмегімен және нормативтердің көмегімен әрбір жұмыстың ұзақтығы бағаланады. Содан кейін тораптық график құрылады (сызылады). Тораптық графикті реттеген соң оқиғалар мен жұмыстардың параметрлері есептеледі, уақыт резервтері мен критикалық жол анықталады. Соңында тораптық графикке талдау жүргізіліп, оңтайландырады, қажет болған жағдайда ол жаңадан қайта сызылып, оқиғалары мен жұмыстарының параметрлері қайта есептеледі.

ТБЖ-да жұмыс жасау термині бірнеше мағынаға ие. Біріншіден, бұл нақты жұмыс – ресурстардың жұмсалуын қажет ететін, уақыт ішінде созылған процесс (мысалы, бұйымды жинау, аспапты сынау және т.б.). Әрбір нақты жұмыс анық, айқын сипатталған болуы тиіс және оның жауапты орындаушысы болуы керек.

Екіншіден, бұл күту – еңбектің, материалдық ресурстардың немесе уақыттың шығындалуын қажет етпейтін, уақыт ішінде созылған процесс (мысалы, бояудан кейінгі кептіру процесі, металдың тозу процесі, бетонның қатаю процесі және т.б.). Ол бір жұмыстың мүмкіншіліктері екінші жұмыстың нәтижелеріне тікелей тәуелді екендігін көрсетеді. Жалған жұмыстың ұзақтығы нөлге тең деп алынады.

Оқиға – бұл қандай да бір процестің аяқталу сәті, ол жоба орындалуының бөлек кезеңін көрсетеді. Оқиға бөлек жұмыстың жеке нәтижесі немесе бірнеше жұмыстың жалпы нәтижесі болуы мүмкін. Оқиға өзіне дейінгі жұмыстардың барлығы аяқталғанда ғана орындалады. Оқиғадан кейінгі жұмыстар оқиға аяқталған кезде ғана бастала алады. Бұл жерден оқиғаның қос мағыналы сипаты келіп шығады: өзінің алдындағы барлық жұмыстар үшін ол ақырғы болып табылады, ал өзінен кейінгі жұмыстар үшін бастапқы болып табылады. Және де бұл жерде оқиға ұзаққа созылмайды және лезде орындалады деп қабылданады. Сондықтан тораптық модельге кіретін әрбір оқиға

толық, анық және жан-жақты анықталған болуы тиіс, оның тұжырымдамасы өзінің алдындағы барлық жұмыстардың нәтижелерін қамтуы тиіс.

Тораптық модель оқиғаларының ішінен бастапқы және қорытынды оқиғаларды ажыратады. Бастапқы оқиғаның алдында модельде көрсетілген жұмыстар кешеніне жататын жұмыстар мен оқиғалар болмайды. Қорытынды оқиғадан кейін жұмыстар мен оқиғалар болмайды.

7.2. Тораптық жоспарлау есептері

Тораптық жоспарлау келесі есептерді шешеді:

- Ағаштарды құру туралы есеп.
- Ең қысқа жол туралы есеп.
- Минимал бағаның максимал ағыны туралы есеп.
- Жұптар мен жабындыларды іздеу есебі.
- Пошташы есебі.
- Коммивояжер есебі.
- Гамильтондық контур туралы есеп.
- Орналастыру есебі.
- Жобаларды басқару есебі.

Келесі әдістердің көмегімен тораптық жоспарлау есептерін шешуге болады:

- ағаштарды құру алгоритмдері;
- ең қысқа және оңтайлы жолдарды іздеу алгоритмдері;
- жұптар мен жабындыларды іздеу алгоритмдері;
- пошташы есебі;
- коммивояжер есебі;
- гамильтондық контурдың бар болу шарты;
- орналастыру есептері, центр мен медианаларды іздеу;
- тораптық график, критикалық жол әдісі;
- Форда-Фалкерсон алгоритмі;
- PERT – алгоритмі.

Осылайша, тораптық және графтық модельдер жүйелерді жобалау кезінде, жұмыстарды жоспарлау кезінде, өнімдерді үлестіру кезінде, транспорттық тасымалдарды ұйымдастыру кезінде, түрлі халыққа қызмет көрсету орталықтарын

орналастыру кезінде және т.с.с кездерде кездесетін кең ауқымды есептер класын қамтиды.

Көптеген практикалық жағдайларда бұл есептер сызықтық мақсатты функциямен және сызықтық шектеулермен сипатталады, сондықтан оларды шешу үшін сызықтық бағдарламалаудың белгілі әдістерін қолдана беруге болатын еді. Алайда мұндай есептерге тән ерекшелік – есеп көлемінің үлкендігі, бұл ерекшелік нақты жүйелердің есептеу ресурстарын үнемдеуге және бастапқы мәліметтердің өзгеруіне байланысты олардың икемділігін қамтамасыз етуге мүмкіндік беретін тиімдірек оңтайландыру алгоритмдерін іздеуді қажет етеді. Мұндай алгоритмдерді тораптар мен графтарда көрсету арқылы құруға болады. Бұл оқулықта графтар теориясының негізгі мәселелерінің инженерлік мазмұны келтірілген. Бұл бөлімнің материалдары түрлі математикалық даярлығы бар мамандарға қолжетімді болып табылады.

Ең қысқа жол туралы және минимал бағаның максимал ағыны туралы есептер берілген бөлімде қарастырылған. Коммивояжер есебі және оны шешу әдістері бұл есептің айрықша ерекшеліктеріне байланысты бөлек тарауда қарастырылған.

7.3. Негізгі ұғымдар мен анықтамалар

Торап түйіндер (төбелер, қосылу нүктелері) мен осы түйіндерді қосатын доғалар (үзбелер, қабырғалар) жиынынан тұрады. Егер доғаның белгілі бір бағыты бар болса, онда ол бағытталған доға деп аталады. Кері жағдайда – бағытталмаған доға деп аталады. Бағытталған және бағытталмаған доғалардың түйіндерін бейнелеу үшін сәйкесінше шеңберлер, бағыты бар және бағыты жоқ сызықтар қолданылады.

Келесідей белгілеулер қолданылады:

N_i – і түйінін белгілеу үшін;

A_{ij} – N_i-дан N_j-ға апаратын бағытталған доғаны белгілеу үшін.

Егер тораптың X түйіндер жиынын $\overline{X}$ ішкі жиынына бөлген кезде A_{ij} немесе A_{ji} доғасы табылатын болса, онда торап байланысқан торап деп аталады, мұндағы $N_i \in X,\ N_j \in \overline{X}$.

Түйіндер мен доғалардың реттілігі тізбек деп аталады. Тұйықталған тізбек цикл деп аталады.

Әрбір алдыңғы доғаның соңы кейінгі доғаның басымен сәйкес келетін доғалар тізбегі жол деп аталады. (Жол бойынша әдетте газ, сұйықтық немесе транспорт жіберіледі).

Торапта екі арнайы түйін ажыратылады: бастапқы түйін – N_s, ақырғы түйін – N_t.

Ағын деп X_{ij} теріс емес сандарының жиынын айтады, олардың әрқайсысы тораптың доғасына сәйкес келеді және келесі шарттарды қанағаттандырады:

$$\left.\begin{array}{r} -V,\ j = S \\ \sum_i x_{ij} - \sum_k x_{jk} = 0,\ j \neq S; t \\ V,\ j = t \end{array}\right\} \qquad (7.1)$$

$$0 \leq x_{ij} \leq d_{ij} \qquad (7.2)$$

мұндағы x_{ij} – A_{ij} доғасымен жүретін ағын.

теріс емес V – ағын шамасы.

(7.1) шектеуі мынаны білдіреді: бастапқы түйін мен ақырғы түйіннен басқа әрбір түйіннен қанша ағын шықса, сонша ағын кіреді (энергияның сақталу шарты), ал (7.2) шарты: доға бойынша жүретін x_{ij} ағыны оның d_{ij} өткізгіштік қабілетімен шектелген екендігін білдіреді.

Тораптық жоспарлау алгоритмдері ең қысқа жол туралы, максимал ағын туралы, тағайындау туралы, сұраныс пен ұсыныс туралы, ресурстарды үлестіру туралы есептерді шешуге мүмкіндік береді.

Мысалы, тораптағы максимал ағын шамасын табу есебі сызықтық бағдарламалау есебі болып табылады, оның мақсатты функциясы:

$$\max V = \sum_j x_{sj} \qquad (7.3)$$

шектеулері: (7.1), (7.2).

Егер A_{ij} доғалары үшін $N_i \in X,\ N_j \in \overline{X}$ болса, немесе керісінше болса, онда A_{ij} доғаларының жиыны $(X,\overline{X})$ кесігі деп аталады.

$(X,\overline{X})$ кесігінің өткізгіштік қабілеті немесе шамасы деп $d(X,\overline{X}) = \sum_{(i,j)} d_{ij}$ шамасын айтамыз, мұндағы қосынды $N_i \in X$ - дан $N_j \in \overline{X}$-ға апаратын және барлық бағытталмаған доғалар бойынша апаратын доғалардан алынады.

N_s-ті N_t-мен қосатын және минимал өткізгіштік қабілетке ие кесік минимал кесік деп аталады.

7.4. Ең қысқа жол туралы есеп

Форд алгоритмі:

1^0. 0 төбесін α_0 индексімен белгілейміз.

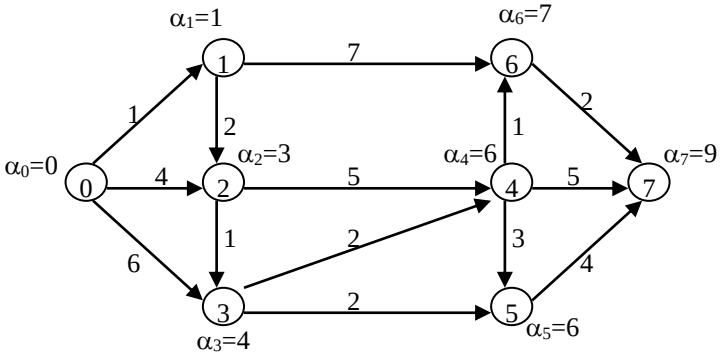
2^0. Нөмірлену реті бойынша тораптық графиктің барлық төбелерін келесі түрде белгілеп шығамыз: $\alpha_j = \min_{i,j} [\alpha_i + l_{ij}]$, мұндағы i,j – төбелердің нөмірлері.

3^0. α_n мәнін – ең қысқа жолдың ұзындығын табамыз.

4^0. Тораптық графиктің соңынан басына қарай жылжи отырып, $\min [\alpha_j - \alpha_i] = l_{ij}$ болатын доғалар ретінде ең қысқа жолға кіретін доғаларды анықтаймыз.

15-мысал.

0 пунктінен n пунктіне баратын барлық мүмкін маршруттардың тораптық графигі белгілі болсын (18-суретті қараңыз). Теріс емес l_{ij} сандары берілген – n_i-дан n_j-ға дейінгі жол ұзындығы. Торап тығыз байланысқан болып саналады, яғни оның кез келген екі төбесінің арасында жол бар деген сөз. 0 пунктінен n пунктіне дейінгі ең қысқа жолды анықтау керек (берілген мысалда n=7).



18-сурет

Шешуі:

1. $\alpha_0=0$

2. $\alpha_1=l_{01}=1$

$\alpha_2=\min[0+4;1+2]=3$

$\alpha_3=\min[0+6;3+1]=4$

$\alpha_4=\min[3+5;4+2]=6$

$\alpha_5=\min[4+2;6+3]=6$

$\alpha_6=\min[6+1;1+7]=7$

$\alpha_7=\min[7+2;6+5;6+4]=9$

3. Сонымен, $\alpha_n=\alpha_7=9$

4. Соңынан басына қарай жылжи отырып, $\min[\alpha_j-\alpha_i] = l_{ij}$ шартынан ең қысқа жолға кіретін доғаларды анықтаймыз

7-ден: $\min[\alpha_7-\alpha_6; \alpha_7-\alpha_4; \alpha_7-\alpha_5]=\min[9-7;9-6;9-6]=2$, 7←6

6-дан: $\min[\alpha_6-\alpha_4; \alpha_6-\alpha_1]=\min[7-6;7-1]=1$, 7←6←4

4-тен: $\min[\alpha_4-\alpha_2; \alpha_4-\alpha_3]=\min[6-3;6-4]=2$, 7←6←4←3

3-тен: $\min[\alpha_3-\alpha_2; \alpha_3-\alpha_0]=\min[4-1;4-0]=3$, 7←6←4←3←2

2-ден: $\min[\alpha_2-\alpha_1; \alpha_2-\alpha_0]=\min[3-1;3-0]=2$, 7←6←4←3←2←1

1-ден: $[\alpha_1-\alpha_0]=1$, 7←6←4←3←2←1←0

Сонымен, ең қысқа жол 0→1→2→3→4→6→7.

Жолдың ұзындығы $\alpha_n=9$.

7.5. Максималды ағын және минималды баға туралы есеп. Форд-Фалкерсон әдісі

Осы параграфтың басында максималды ағын мен минималды баға туралы есепті шешу әдісі негізделетін екі теореманың тұжырымдамасы келтірілген.

Теорема 4.

Егер c_{ij} матрицасы бар есептің оңтайлы шешімі бар болса, онда бұл шешім $c_{ij}'=c_{ij}+\delta,t(i,j)$ түрлендірілген матрицасы бар есеп үшін де оңтайлы болады.

Теорема 5.

Түрлендірілген есептің ізделініп отырған шешімі оңтайлы болуы үшін келесі оңтайландыру шарттарының орындалуы қажетті және жеткілікті:

1) $c_{ij}>0$, $x_{ij}=0$;
2) $c_{ij}<0$, $x_{ij}=d_{ij;}$
3) $c_{ij}=0$, $0\le x_{ij}\le d_{i.j.}$

Форд-Фалкерсон алгоритмі:

1^0. S және t бастапқы және соңғы түйіндерін қосайық, және мұндағы $a_s = \sum_{j\in S} a_{ij}$ және $b_t = \sum_{j\in T} b_j$.

2^0. x_{ij} бастапқы ағынын барлық доғалар бойынша 0-ге тең деп аламыз.

3^0. $S=[-, a_s]$ төбесін белгілейміз.

4^0. t төбесіне жетуге талпына отырып, S-пен көршілес төбелер үшін белгілеулерді орналастырып шығамыз.

Орналастыру ережелері мынадай:

j түйіні і түйінінен белгілене алады, егер:

а) $c_{ij}'\le 0$, $x_{ij}<d_{ij}$

б) $c_{ij}'\ge 0$, $x_{ij}>0$

Егер і түйінінде $i=[k^{\pm},\varepsilon_i]$ белгісі болса, онда j түйіні келесі белгіні алады:

$j=[i^+, \varepsilon_j=\min(\varepsilon_i, d_{ij}-x_{ij})]$ (а) жағдайында,

$j=[i^-, \varepsilon_j=\min(\varepsilon_i, x_{ij})]$ (б) жағдайында.

± таңбалары белгілеу тік доға бойынша өтіп жатыр ма, әлде кері доға бойынша өтіп жатыр ма, соған байланысты қойылады, яғни (а) жағдайында «+» қойылады, (б) жағдайында «-» қойылады. Соңғы төбе белгіленгенше 4^0 қадамды жалғастыра береміз.

Егер соңғы түйінге жеткен болсақ, онда 5^0 қадамға көшеміз, егер жетпеген болсақ, 6^0 қадамға көшеміз.

5^0. Егер соңғы түйінге жеткен болсақ, яғни t түйіні белгіленген болса, яғни s-тан t-ға дейінгі жол табылған болса, онда ағындарды келесідей өзгертеміз:

$$\begin{cases} x_{ij}+\varepsilon_t, (i,j)\in \text{тік доғаға;} \\ x_{ij}'= x_{ij}-\varepsilon_t, (i,j)\in \text{кері доғаға;} \\ x_{ij}, \text{қалған жағдайларда.} \end{cases}$$

Сондай-ақ $a_s'=a_s-\varepsilon_t$ мәнін де өзгертеміз. 3^0 қадамға көшеміз.

6^0. Егер соңғы түйінге жетпеген болсақ, онда доғалардың бағасын өзгертеміз, ол үшін келесі жиындарды бөліп аламыз:

$A_1=\{(i,j)/i\in X, j\in \overline{X}, c_{ij}\geq 0\}$

$A_2=\{(j,i)/i\in X, j\in \overline{X}, c_{ij}<0\}$

мұндағы X – белгіленген төбелер жиыны, $\overline{X}$ – белгіленбеген төбелер жиыны.

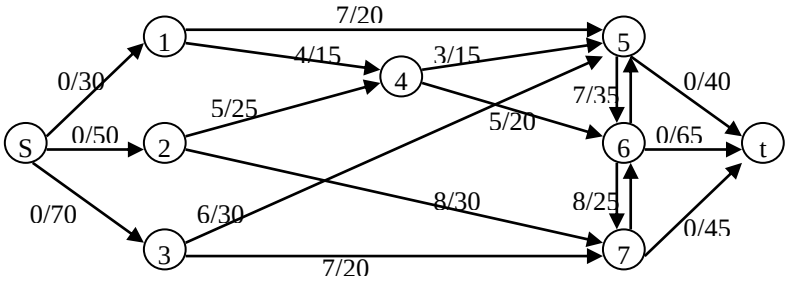
Сонда $\delta_1=\min_{A1} c_{ij}$, $\delta_2=\min_{A2}|c_{ij}|$, $\delta=\min(\delta_1, \delta_2)$.

$$\begin{cases} c_{ij}-\delta, i\in X, j\in \overline{X}; \\ c_{ij}'= \quad c_{ij}+\delta, i\in \overline{X}, j\in X; \\ c_{ij}, \text{қалған жағдайларда.} \end{cases}$$

Осылайша, A1-ден немесе A2-ден кем дегенде бір мүмкін доға қарастырылады. 3^0 қадамға көшеміз.

7^0. $a_s=0$ болғанда, немесе A_1 мен A_2 0-ге тең болғанда, яғни минимал кесік алынғанда, есептеулер аяқталады.

16-мысал



19-сурет

1) s және t төбелерін енгіземіз.

$a_s=\sum a_j=30+50+70=150$

2) $x_{ij}=0\ \forall(i,j)$ деп аламыз.

3) $s=[-, \varepsilon_s=a_s=150]$ төбесін белгілейміз.

S-тен 1, 2, 3-ті белгілеуге болады, себебі $c_{sj}=0$:

$1=[S^+,\varepsilon_1=\min(\varepsilon s, d_{s1}-x_{s1})=\min(150,30-0)=30]$

$2=[S^+,\varepsilon_2=\min(\varepsilon s, d_{s2}-x_{s2})=\min(150,50-0)=50]$

$3=[S^+,\varepsilon_3=\min(\varepsilon s, d_{s3}-x_{s3})=\min(150,70-0)=70]$

Басқа төбелерді белгілей алмаймыз, себебі $c_{ij}\geq 0$, $x_{ij}>0$ шарттары орындалмайды (соңғысы 0-ге тең).

6) Соңғы түйінге жетпедік, сондықтан келесі жиындарды енгіземіз:

$A_1=\{\forall c_{ij}>0$: (1,5), (1,4), (2,4), (2,7), (3,5), (3,7)$\}$

$A_2=0$, себебі $c_{ij}<0$ болатындай (i,j) жоқ.

A_1, A_2-ге кіретін доғалардың бағасы:

$\delta_1=\min(7,4,5,8,6,7)=4$, $\delta_2=\infty$, $\delta=\min(\delta_1, \delta_2)=4$.

A_1 жиыны бойынша доғалардың бағасы өзгертеміз (суретте көрсетілген). Енді 4-ші төбені белгілеуге болады.

$4=[1^+,\varepsilon_4=\min(\varepsilon_1, d_{14}-x_{14})=\min(80,15-0)=15]$

6) Тағы да соңғы түйінге жетпедік, себебі 4-ші төбеден t төбесін белгілей алмаймыз:

$A_1=\{(1,5), (4,5), (4,6), (2,7), (3,5), (3,7)\}$. Мұнда (2,4) доғасы қарастырылмайды (Неге екенін айтыңыз?).

$A_2=0$.

δ_1=min(3,3,5,4,2,3)=2, δ=4.

A_1 жиынының барлық доғаларының бағасы c_{ij}-2 болып өзгереді (суретте көрсетілген).

Келесі төбелерді белгілеуге болады:

5=[3^+,ε_5=min(ε_3, d_{35}-x_{35})=min(70,30-0)=30]

t=[5^+,ε_t=min(ε_5, d_{5t}-x_{5t})=min(30,40-0)=30]

4) Соңғы түйінге жеттік, яғни S→3→5→t жолы табылды, осы жол арқылы тауардың 30 бірлігі жіберіледі. Осы доғалар бойынша ағындарды 30 бірлікке ауыстырамыз, (3,5) доғасы толып кетті, бұдан ары қарай ол қарастырылмайды.

5) a_s=150-30=120. s төбесінен бастап төбелерді жаңадан белгілейміз:

S=[-,120]

1=[S^+,ε_1=min (120,30-0)=30],

2=[S^+,ε_2=min (120,50-0)=50],

3=[S^+,ε_3=min (120,70-30)=40],

4=[S^+,ε_4=min (30,15-0)=15]

6) Соңғы түйінге жетпедік, сондықтан екі жиын енгіземіз:

A_1={(1,5), (4,5), (4,6), (2,7), (3,7)}.

A_2=0.

δ_1=min(1,1,3,2,1)=1, δ=1.

Жиындарға кіретін доғалардың бағасын өзгертеміз де, есептеу процесін жалғастырамыз.

A_1 және A_2 жиындары бос болғанша (берілген мысал үшін) алгоритмнің барлық қадамдарын рет-ретімен орындап шығамыз.

Есептің оңтайлы шешімі:

s→3→5→t (30 бірлік)

s→1→5→t (10 бірлік)

s→3→7→t (20 бірлік)

s→2→7→t (25 бірлік)

s→1→4→6→t (15 бірлік)

s→2→4→6→t (5 бірлік)

s→1→5→6→t (5 бірлік)

s→2→4→5→6→t (15 бірлік)

Қарастырылған есепте s бастапқы түйінен t соңғы түйініне дейін барлық доғалармен жүктің 150 бірлігі тасымалдануы тиіс. Жалпы алғанда, көрсетілген жүктің 125 бірлігі тасымалданды.

(3,5), (3,7), (7,t), (1,4), (4,6), (5,t), (s,1), (4,5) доғалары қанықкан доғалар болды. Бұл дегеніміз – берілген доғалар бойынша жүретін ағын олардың өткізгіштік қабілетіне тең деген сөз.

Форд-Фалкерсон есептеу алгоритмін қолдану кезінде 11 итерация жүргізілді. (Өздігінен тексеріңіздер).

7.6. Тораптық графиктердің жоспарлы параметрлерін есептеу

Тораптық модельдер өндірістік жүйенің ұйымдастырушылық құрылымы бойынша да, тораптық графиктердің тағайындалуы бойынша да әр түрлі болуы мүмкін, сондай-ақ оларды нормативтік мәліметтер мен ақпаратты өңдеу құралдары қолдануы мүмкін. Ұйымдастырушылық құрылым бойынша тораптық жоспарлаудың фирма ішілік және салалық модельдері ажыратылады, тағайындалуы бойынша бір ретті және тұрақты әрекет етуші болып ажыратылады. Тораптық модельдер детерминдік, ықтималдық және аралас болады. Детерминдік тораптық графиктерде стратегиялық жобаның барлық жұмыстары, олардың ұзақтығы мен өзара байланысы, сондай-ақ күтілетін нәтижеге қойылатын талаптары алдын ала белгілі болады. Ықтималдық модельдерде көп процестер кездейсоқ сипатқа ие. Аралас тораптарда жұмыстардың бір бөлігі анықталған, ал екіншісі анықталмаған болады. Модельдер, сонымен қатар, бір мақсатты және көп мақсатты болуы мүмкін.

Тораптық графиктерді тұрғызу кезінде әрбір кәсіпорындағы шынайы жағдайлар мен жұмыстардың нақты сипаттамаларын ескеру қажет.

Тораптық модельдер барлық отандық кәсіпорындарда ұзақ мерзімді жоспарларды, сонымен қатар ағымдық жоспарларды жасау кезінде кеңінен қолданылуы мүмкін. Тораптық жоспарлау түрлі өндірістік ресурстардың болашақтағы қажеттілігін анықтап қана қоймай, олардың қазіргі кездегі тиімді пайдаланылуын реттеуге мүмкіндік береді. Тораптық графиктердің көмегімен өндірістің барлық материалдық, еңбектік, қаржылық және көптеген басқа да ресурстары мен

құралдарын идеалды (жоспарланатын) және шынайы (реалды) экономикалық жағдайларда бірыңғай жүйеге біріктіруге болады.

Біздің кәсіпорындарда тораптық жоспарлау және экономикалық қызметті басқару жүйелерін құру, ең алдымен жоспарлы органдардың құрылымы мен функцияларын анықтауды, жоспарлау мақсатын негіздеуді және жоспарлау объектісін таңдауды, жобаны тораптық моделін құруды, бастапқы жоспарлау және жобаны жедел басқару кезеңдерінде модельдің жұмыс жасау ретін орнатуды көздейді. Нақты жағдайларға байланысты, тораптық модельдерді жасау процесіндегі негізгі басқарушылық және орындаушылық қызметтерді кәсіпорынның барлық деңгейдегі жоспарлы қызметтердің басшылары және түрлі категориядағы экономист-менеджерлер орындай алады. Жоспарлаушы мамандар тораптық жоспарлардың жасалуын тікелей қамтамасыз етеді және олардың орындалу барысын бақылап отырады. Жоба жетекшілері немесе жауапты орындаушылар әдетте кәсіпорынның тиісті жоспарлы бөлімшелерінің құрамынан тағайындалады.

Барлық тораптық жоспарлау жүйелерінде негізгі модельдеу объектісі алдағы жұмыстардың әр түрлі кешендері болып табылады, мысалы маркетингтік зерттеулер, жобалық әзірлемелер, жаңа тауар өндірісін игеру және басқа да жоспарлы іс-шаралар. Жоспарланатын жұмыстар кешенінің мазмұны мен орындалу мерзімдері сан алуан болуы мүмкін – 10-15 операцияны қамтитын қарапайым есептік-техникалықтан бастап, бірнеше мың іс-шараны көздейтін күрделі құрылыстық-монтаждық болуы мүмкін. Осындай жұмыс кешенінің кез келген жүйесінің ортақ қасиеттері – оларды бөлек процестер жиыны ретінде қарастыру мүмкіндігі, прогрессивтік технологиялық әдістерді қолдану қажеттілігі, ақырғы нәтижелерге қол жеткізу үшін ортақ мақсаттардың болуы және т.с.с.

Әр түрі өндірістік жүйелерді немесе басқа да экономикалық объектілерді тораптық жоспарлаудың аса маңызды кезеңдері мыналар:

– жұмыстар кешенін жеке бөліктерге бөлшектеу және оларға жауапты орындаушыларды тағайындау;

– әрбір орындаушының алға қойылған мақсатқа жетуге қажетті оқиғалар мен жұмыстарды анықтауы және сипаттауы;

– бастапқы тораптық графиктерді тұрғызу және жоспарланатын жұмыстардың мазмұнын нақтылау;

– жеке тораптарды біріктіру және жұмыстар кешенінің орындалуының жиынтық тораптық графигін тұрғызу;

– тораптық графиктегі әрбір жұмыстың орындалу уақытын негіздеу немесе нақтылау.

Жоспарланатын жұмыстар кешенін бөлшектеуді жоба жетекшісі жүзеге асырады. Тораптық жоспарлау барысында орындалатын жұмыстарды үлестірудің екі тәсілі қолданылады: орындаушылар арасында функцияларды горизонталь үлестіру және жоба жетекшілігінің деңгейлерінің сұлбасын вертикаль құру. Бірінші жағдайда қарапайым жүйе немесе объект жеке процестерге, бөліктерге немесе элементтерге бөлінеді, ол үшін ұлғайтылған тораптық график құрылуы мүмкін. Сосын әрбір процесс операцияларға, амалдарға және басқа да есептеу әрекеттеріне бөлінеді. Жұмыстар кешенінің әрбір құрамдасына өзіндік тораптық график құрылады. Екінші жағдайда жобаланатын күрделі объект жобаны басқарудың тиісті деңгейлерінің белгілі иерархиялық құрылымын құрудың көмегімен жеке бөліктерге бөлінеді. Әрбір деңгейде тораптық графиктерді олардың жетекшілері немесе жауапты орындаушылары құрады. Олардың әрқайсысы тораптық жоспарлау процесінде келесідей қызметтерді атқарады:

– бекітілген жұмыс көлеміне бастапқы тораптық график құрады;

– өзіне бекітілген жұмыстардың орындалу барысын бағалайды да, өзінің басшысына қажетті ақпаратты ұсынады;

– өндірістік бөлімшелер мен функционалды органдардың қызметкерлерімен бірге жоспарлы және басқарушылық шешімдерді дайындауға қатысады;

– тиісті басқару органы қабылдаған шешімдердің барлығын орындайды.

Тораптық модельдердің негізгі параметрлері бөлек процестердің, сондай-ақ бүкіл жұмыстар кешенінің орындалуының жоспарланатын бағалық және уақыттық көрсеткіштері болып табылады. Тораптық графикте көзделген әрбір жұмыс өзінің орындалуы үшін жұмыс уақытының, материалдық, еңбектік, қаржылық және басқа да өндірістік ресурстардың белгілі бір шығынын талап етеді. Тораптық модельдердің уақыттық және бағалық сипаттамалары бүкіл жұмыстар кешенін немесе процестерді орындауға қажетті экономикалық ресурстарды жұмсаудың ең маңызды жалпылаушы көрсеткіштері болып табылады. Кәсіпорындағы стратегиялық жоспарлаудың және өндірістік қызметті басқарудың тораптық жүйелерінің көпшілігі үшін ең алдымен нақты ресурстардың қажеттілігі туралы мәліметтер кәдімгі түрде көрсетілуі қажет. Тораптық жоспарлауда қолданылатын барлық ресурстар екі түрге бөлінеді – қатталатын және қатталмайтын.

Қатталатын, немесе жаңартылмайтын өндірістік ресурстарға мына шикізаттар жатады: материалдар, жартылай фабрикаттар, дайын тауарлар, жанармай және басқа да айналмалы құралдар. Оларға, сондай-ақ, ақшалай және бағалық ресурстарды да жатқызуға болады, сондықтан да бағаны қатталатын ресурстардың бір түрі ретінде қарастыруға болады. Алайда тораптық жоспарлауда баға орындалатын жұмыстар кешенінің жалпы экономикалық сипаттамасы ретінде жүретін модельдер айрықша қолданылады. Қатталатын ресурстар тораптық графиктерде жоспарланатын жұмыстарды орындау процесінде тікелей жұмсалады және қайта қолданылмайды. Мұндай ресурстар дер кезінде қолданылмаған жағдайда келесі жұмыстарда қолданылуы мүмкін. Әдетте қолданылмайтын қоймалық ресурстардың саны мен бағасы өзгеріссіз қалады деп ұйғарылады, дегенмен ұзақ уақытқа модельдеу кезінде ресурстардың тек сандық қана емес, сапалық көрсеткіштерінің де төмендеуін ескерген жөн.

Қатталмайтын, немесе жаңартылмайтын ресурстарға жұмыс күші, өндіріс құралдары, жұмыс құралдары, өндірістік алаң және басқа да негізгі қорлар жатады. Мұндай ресурстар

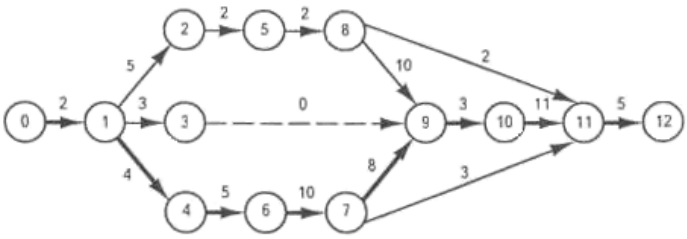
жұмыс процесінде тиімді пайдаланынуы тиіс. Ұзақ уақытқа модельдеу кезінде қатталмайтын ресурстардың бастапқы бағасының өзгеруін ескеру қажет, мысалы, технологиялық жабдық өнімділігінің төмендеуі, қызметкерлер құрамының кәсіби біліктілігінің артуы және т.с.с. Қысқа мерзімдік тораптық модельдерде жоспарланған технологиялық процестер мен жұмыстардың орындалуына қажетті қатталмайтын ресурстарға деген қажеттілік әдетте тұрақты деп қабылданады.

Демек, тораптық модельдердегі жоспарланатын негізгі параметрлерге жұмыстардың орындалу ұзақтығы, критикалық жол, оқиғалардың орындалу уақытының резервтері және т.б. сияқты уақыттық көрсеткіштер жатады.

Кез келген тораптық графиктің ең маңызды параметрі критикалық жол болып табылады. Тораптық графиктегі жол деп бірнеше оқиғаны өзара байланыстыратын кез келген жұмыстар (бағдарлар) тізбегін айтамыз.

Тораптың бастапқы және соңғы оқиғаларын байланыстыратын жол толық жол болып саналады, ал басқалары толық емес болып саналады. Әрбір жол өзінің ұзақтығымен сипатталады, ол оны құрайтын жұмыстардың ұзақтығының қосындысына тең болады.

Ұзақтығы ең үлкен болатын толық жол критикалық жол деп аталады. Осылайша, критикалық жол – бұл бастапқы оқиғадан соңғы оқиғаға алып келетін, уақыт бойынша ең ұзаққа созылған жұмыстар тізбегі. Тораптық графиктегі (20-сурет) критикалық жол 0 – 1 – 4 – 6 – 7 – 9 – 10 – 11 – 12 нөмірлерімен белгіленген оқиғалар мен жұмыстар тізбегі арқылы өтеді, және 48 адам-күнге тең.



20-сурет. Жобаны орындаудың тораптық графигі

Жобалық жұмыстарды орындаудың графигі үшін оқиғалардың орындалуының ерте және кеш мерзімдерін, сондай-ақ уақыт резервтерін ағымдық ережелер бойынша есептеу қажет болып танылады (20-сурет).

Оқиғалардың орындалуының ерте мерзімдерінің есептелуі бастапқыдан соңғысына қарай түзу ретпен орындалады.

$$T_{P_0} = 0;$$
$$T_{P_1} = t_{0-1} = 2;$$
$$T_{P_2} = t_{0-2} = 2 + 5 = 7;$$
$$T_{P_3} = t_{0-1-3} = 2 + 3 = 5;$$
$$T_{P_4} = t_{0-1-4} = 2 + 4 = 6;$$
$$T_{P_5} = t_{0-1-2-5} = 2 + 5 + 2 = 9;$$
$$T_{P_6} = t_{0-1-4-6} = 2 + 4 + 5 = 11;$$
$$T_{P_7} = t_{0-1-4-6-7} = 2 + 4 + 5 + 10 = 21;$$
$$T_{P_8} = t_{0-1-2-5-8} = 2 + 5 + 2 + 2 = 11;$$
$$T_{P_9} = t_{max}(l_1 = 11;\ l_2 = 5;\ l_3 = 29) = 29;$$
$$T_{P_{10}} = T_{P_9} + t_{9-10} = 29 + 3 = 32;$$
$$T_{P_{11}} = t_{max}(T_8 + t_{8-11} = 13;\ T_{10} + t_{10-11} = 43;\ T_7 + t_{7-11} = 24) = 43;$$
$$T_{P_{12}} = T_{P_{11}} + t_{11-12} = 43 + 5 = 48.$$

12-ші оқиғаның орындалуының ерте мерзімі тораптық графиктің критикалық жолына сәйкес келеді: $L_{кр} = 48$ күн.

Қалған толық жолдар мынаған тең:

$$L_{0-1-2-5-8-11-12} = 18;$$
$$L_{0-1-2-5-8-9-10-11-12} = 40;$$
$$L_{0-1-2-3-9-10-11-12} = 24;$$
$$L_{0-1-4-6-7-11-12} = 29.$$

Оқиғалардың орындалуының кеш мерзімдерінің есептелуі соңғысынан бастапқысына қарай кері ретпен орындалады.

$$T_{П_{12}} = L_{кр} = 48;$$
$$T_{П_{11}} = L_{кр} - t_{11-12} = 48 - 5 = 43;$$
$$T_{П_{10}} = L_{кр} - t_{10-12} = 48 - 16 = 32;$$
$$T_{П_{9}} = L_{кр} - t_{9-12} = 48 - 19 = 29;$$
$$T_{П_{8}} = L_{кр} - l_{max}(l_{8-12} = 7;\ l_{8-9-10-11-12} = 29) = 48 - 29 = 19;$$
$$T_{П_{7}} = L_{кр} - l_{max}(l_{7-9-10-11-12} = 27;\ l_{7-11-12} = 8) = 48 - 27 = 21;$$
$$T_{П_{6}} = L_{кр} - t_{6-12} = 48 - 37 = 11;$$
$$T_{П_{5}} = L_{кр} - t_{5-12} = 48 - 31 = 17;$$
$$T_{П_{4}} = L_{кр} - t_{4-12} = 48 - 42 = 6;$$
$$T_{П_{3}} = L_{кр} - t_{3-12} = 48 - 19 = 29;$$
$$T_{П_{2}} = L_{кр} - t_{2-12} = 48 - 33 = 15;$$
$$T_{П_{1}} = L_{кр} - l_{max}(l_1 = 18;\ l_2 = 22;\ l_3 = 46) = 48 - 46 = 2;$$
$$T_{П_{0}} = 0.$$

Бөлек оқиғалардың орындалуының уақыттық резервтері оларды орындаудың кеш және ерте мерзімдерінің айырмасы болып табылады:

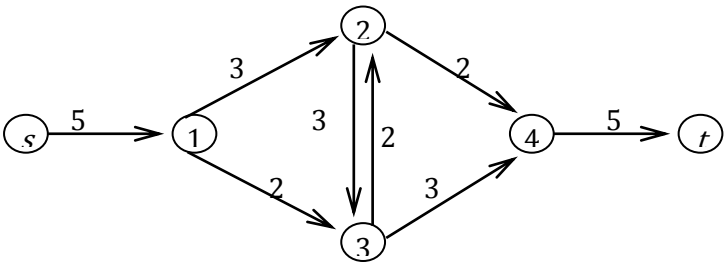
$$R_0 = T_{П_0} - T_{Р_0} = 0 - 0 = 0;$$
$$R_1 = T_{П_1} - T_{Р_1} = 2 - 2 = 0;$$
$$R_2 = T_{П_2} - T_{Р_2} = 15 - 7 = 8;$$
$$R_3 = T_{П_3} - T_{Р_3} = 29 - 5 = 24;$$
$$R_4 = T_{П_4} - T_{Р_4} = 6 - 6 = 0;$$
$$R_5 = T_{П_5} - T_{Р_5} = 17 - 9 = 8;$$
$$R_6 = T_{П_6} - T_{Р_6} = 11 - 11 = 0;$$
$$R_7 = T_{П_7} - T_{Р_7} = 21 - 21 = 0;$$

$$R_8 = T_{П_8} - T_{Р_8} = 19 - 11 = 8;$$
$$R_9 = T_{П_9} - T_{Р_9} = 29 - 29 = 0;$$
$$R_{10} = T_{П_{10}} - T_{Р_{10}} = 32 - 32 = 0;$$
$$R_{11} = T_{П_{11}} - T_{Р_{11}} = 43 - 43 = 0;$$
$$R_{12} = T_{П_{12}} - T_{Р_{12}} = 48 - 48 = 0.$$

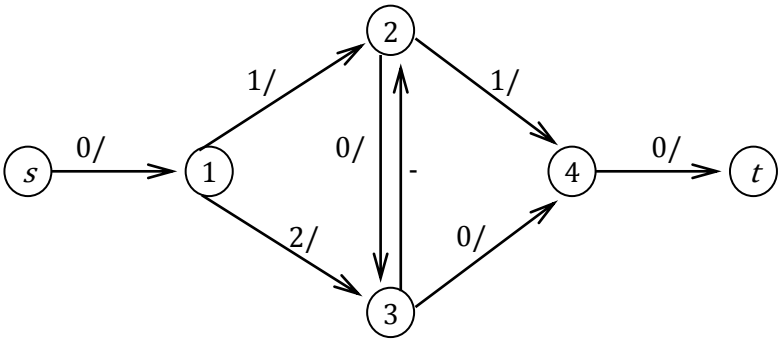
Уақыт резервтерін есептеу арқылы критикалық жолдың тораптық графикте 0 – 1 – 4 – 6 – 7 – 9 – 10 – 11 – 12 оқиғалары арқылы өтетіндігі расталады, уақыт резервтерінің мәндері нөлге тең.

Сұрақтар мен тапсырмалар

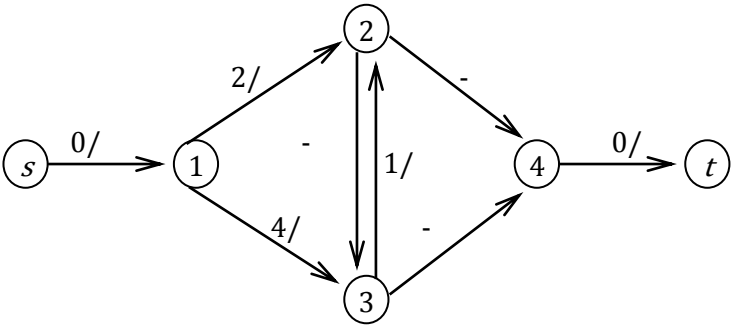
1. Тораптық жоспарлаудағы жол, тізбек, цикл, кесік ұғымдарының анықтамаларын келтіріңіз.
2. Кесіктің өткізгіштік қабілеті қалай анықталады?
3. Қандай типті кесік минималды кесік деп аталады?
4. Максималды ағын мен минимал кесік туралы теореманы тұжырымдаңыз.
5. Қандай жағдайда торап байланысқан торап деп аталады?
6. Максималды ағын мен минималды баға туралы есептің қойылымын келтіріңіз.
7. Тораптық жоспарлау есептерін атап шығыңыз.
8. Тораптың арнайы екі түйінін атаңыз.
9. Тораптық графикте қандай доғалар бар?
10. Келесі тораптық графиктегі ең қысқа жолды табыңыз:



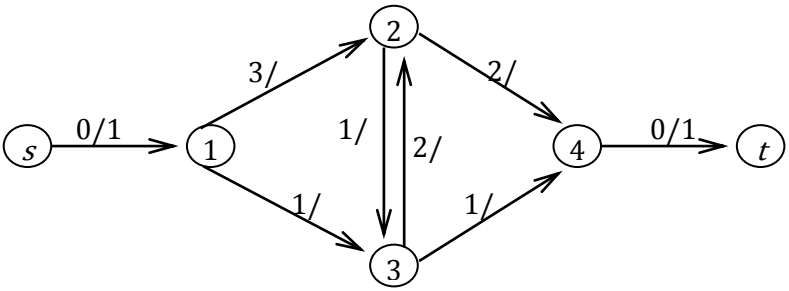
11. Есепті Форд-Фалкерсон әдісімен шешіңіз:



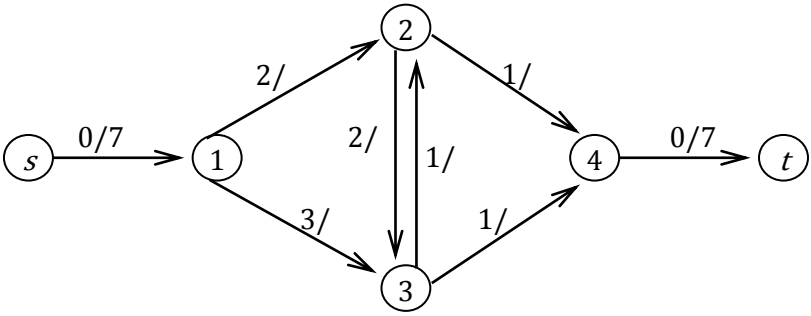
12. Есепті Форд-Фалкерсон әдісімен шешіңіз:



13. Есепті Форд-Фалкерсон әдісімен шешіңіз:



14. Есепті Форд-Фалкерсон әдісімен шешіңіз:



8. КОММИВОЯЖЕР ЕСЕБІ

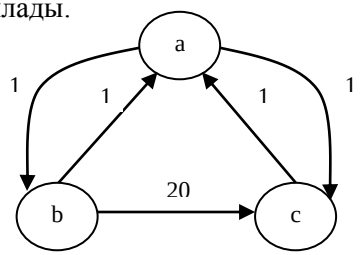
Коммивояжер нақты қызмет көрсету аймағындағы әрбір қалаға барып, үйіне қайтуы керек. Әрине, коммивояжер клиенттерді аралау кезінде жолы мүмкіндігінше қысқа болатындай маршрутты таңдағысы келеді. Осылайша, коммивояжер ұзындығы, ұзақтығы және құны минимал болатын маршрутты іздеу есебін шешуі керек.

Коммивояжер есебін граф есебі ретінде тұжырымдауға болады: G(X,A) графын тұрғызу керек, оның төбелері коммивояжердің аймағындағы қалаларға сәйкес келеді, ал доғалары қала жұптарын байланыстырушы коммуникацияларды көрсетеді. Әрбір (x,y) Є A доғасының l(x,y)≥ 0 ұзындығы қашықтығына, құнына және уақытына тең болсын. G графының әрбір төбесін ең болмаса бір рет қамтитын контур гамильтон контуры деп (осы есепті алғаш болып зерттеген математик В.Р.Гамильтонның атымен) аталады.

Жалпы коммивояжер есебі деп жалпы ұзындығы ең қысқа болатын маршрутты іздеу есебін айтамыз.

Коммивояжер есебі деп жалпы ұзындығы ең қысқа болатын гамильтон контурын іздеу есебін айтамыз.

G графы берілсін, оның әрбір төбесі қалаға сәйкес келеді, ал (x,y) төбелерінің әрбір жұбы ұзындығы l(x,y) болатын доғамен байланысады. Ең қысқа ұзындыққа ие коммивояжер контуры коммивояжердың оңтайлы маршруты деп аталады және жалпы коммивояжересебінің оңтайлы шешімі болып табылады. Ең қысқа ұзындықты гамильтон контуры оңтайлы гамильтон контуры деп аталады және коммивояжер есебінің оңтайлы шешімі болып табылады.



21-сурет. Коммивояжердің оңтайлы жолы

Коммивояжердің оңтайлы жолы міндетті түрде гамильтон контуры болып табылмайды. 21-суретте көрсетілген графты қарастырайық. Бұл графта жалпы ұзындығы 22-ге тең болатын тек бір ғана (a,b), (b,c), (c,a) гамильтон контуры бар. Коммивояжердің (a,b), (b,a), (a,c), (c,a) оңтайлы жолы әрбір төбеден екі рет өтеді және оның жалпы ұзындығы 1+1+1+1=4.

Онда, қандай жағдайларда гамильтон контуры жалпы коммивояжер есебінің шешімі болады?

Теорема. Егер G графының әрбір (x,y) жұбы үшін төмендегі шарт орындалатын болса

$l(x,y) \le l(x,z) + l(z,y)$, барлығы үшін $z \neq x$, $z \neq y$ (8.1)

онда гамильтон контуры G графындағы жалпы коммивояжер есебінің шешімі болып табылады (егер шешім бар болса).

(8.1) шарты «үшбұрыш теңсіздігі» деп аталады және ол x-тен y-ке дейінгі қашықтық кез келген басқа z төбесі арқылы өтетін x-тан y-ке дейінгі қашықтықтан аспайтынын білдіреді.

Дәлелдеу. Гамильтон контуры болып табылатын жалпы коммивояжер есебінің оңтайлы шешімі жоқ деп алайық. L – коммивояжердің оңтайлы маршруты болсын. L гамильтон контуры болмағандықтан, онда қандай да бір z төбесі L маршрутында ең болмағанда екі рет қайталансын. Бірінші рет коммивояжер z төбесіне x төбесінен келіп, одан y төбесіне қарай шықты делік. L маршрутын былайша өзгертейік: коммивояжер x төбесінен шығып, z төбесінен өтіп, тура y төбесіне барады. Осының нәтижесінде алынған L' маршруты да коммивояжер маршруты болып табылады, себебі ол ондағы әрбір төбеге ең болмағанда бір рет барады. Сондай-ақ, (8.1) шартына сәйкес L' маршрутының жалпы ұзындығы L-ң ұзындығынан аспайды. L-ді L'-қа ауыстыра отырып, және осы процедураны қайталай отырып, басқа L"маршрутын аламыз және т.б. Ақырында бұл процесс бізді оңтайлы маршрутқа алып келеді, ол гамильтон контуры болып табылады, себебі әрбір келесі маршрут доғаларының саны өзінің алдындағы маршруттағыдан 1-ге кем, біздің дәлелдеу керегіміз де осы.

Теоремадан шығатыны: егер G графы үшбұрыш теңсіздігін қанағаттандыратын болса, онда G графындағы коммивояжер

есебінің оңтайлы шешімі осы графтағы жалпы коммивояжер есебінің оңтайлы шешімі болып табылады.

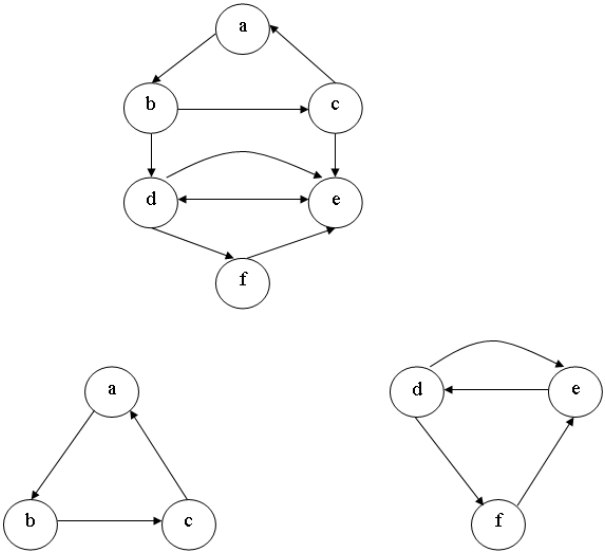
Коммивояжер есебі мен жалпы коммивояжер есебін шешу үшін екі түрлі әдісті қолданбаудың қарапайым тәсілі бар. Егер G графы үшбұрыш теңсіздігін қанағаттандырмаса, онда осы теңсіздікті қанағаттандырмайтын әрбір (x,y) доғасының l(x,y) ұзындығын x төбесінен y төбесіне апаратын ең қысқа жолдың ұзындығымен алмастыру қажет.

Енді x-тен y-ке апаратын доға ұзындығы коммивояжердің x төбесінен y төбесіне қарай тікелей x-тен y-ке емес, осы төбелерді байланыстыратын ең қысқа жолдың бойымен баруын көрсетеді және жолдың ұзындығы үшбұрыш теңсіздігін қанағаттандырады. Егер G графындағы коммивояжер есебінің оңтайлы шешімі ұзындығы жоғарыда сипатталған тәсілмен қысқартылған қандай да бір (x,y) доғасын қамтитын болса, онда оңтайлы шешімде бұл доғаны x пен y төбелері арасындағы ең қысқа жолға кіретін доғалармен алмастыру керек. Мысалы, 21-суретте l(b,c)=20> l(b,a)> l(a,c)=1+1. Бұл жерде (b,c) доғасы үшін (8.1) шарты орындалмайтындығы көрініп тұр. Егер (b,c) доғасының ұзындығын 2-ге дейін азайтсақ (b-дан c-ға дейінгі ең қысқа жол ұзындығы), онда алынған графта тек бір ғана (a,b),(b,c),(c,a) гамильтон контуры пайда болады, оның ұзындығы 1+2+1=4. (b,c) доғасын (b,a) және (a,c) доғаларымен алмастыра отырып, (a,b),(b,a),(a,c),(c,a) маршрутын аламыз, ол бастапқы графтағы коммивояжердің оңтайлы маршруты болып табылады.

Сонымен, коммивояжер есебінің шешімі гамильтон контуры болып табылады. Алайда, гамильтон контуры барлық графтарда болмайды. Демек, оңтайлы гамильтон контуры іздеместен бұрын, оның осы графта бар-жоқтығын тексеру керек. Келесідей анықтамаларды қолданайық. Графта кез келген екі x және y төбелері үшін x-тен y-ке қарай жол бар болса, онда граф тығыз байланысқан деп аталады. Қандай да бір графтың кез келген x$\in$ x_i және y$\in$ x_i төбелер жұбы үшін x-тен y-ке қарай жол бар болса және x_i осындай қасиеттерге ие басқа ешқандай төбелер жиынының ішкі жиыны болмаса, онда x_i ішкі жиыны тығыз байланысқан деп аталады.

Тығыз байланысқан төбелер ішкі жиынынан пайда болған ішкі граф бастапқы графтың тығыз байланысқан компонентасы болып табылады. Мысалы, 21-суретте көрсетілген граф тығыз байланысқан граф деп аталады, себебі бұл графтың әрбір төбесінен кез келген басқа төбесіне қарай жол бар. Осыны өздеріңіз тексеріп көріңіздер.

22-суретте көрсетілген графты қарастырайық. Бұл граф тығыз байланысқан граф деп аталмайды, себебі оның ішіндегі d төбесінен b төбесіне апаратын жол жоқ, дегенмен d-дан b-ға апаратын тізбек бар, ол (d,b) доғасы болып табылады. {a,b,c} төбелері тығыз байланысқан ішкі жиынды түзеді, себебі осы төбелердің әрқайсысынан осыжиындағы кез келген төбеге дейін жол бар. Сондай-ақ, бір де бір қосымша төбе оған осы қасиетінен арылмай қосыла алмайды. Мысалы, d төбесі қосыла алмайды, себебі d төбесінен a төбесіне апаратын жол жоқ. {a,b,c} ішкі жиынынан пайда болған ішкі гарф 22-суретте келтірілген. Бұл граф бастапқы графтың тығыз байланысқан компонентасы болып табылады.



22-сурет. Граф және оның тығыз байланысқан компоненталары

{d,e} төбелер ішкі жиынын қарастырайық. Графта d-дан e-ге қарай және e-ден d-ға қарай жол бар. Алайда, {d,e} тығыз байланысқан ішкі жиын болып табылмайды, себебі оған тығыз байланысу қасиетін жоғалтпастан f төбесі қосыла алады. Сондай-ақ, {d,e,f} жиынына осы қасиетін жоғалтпай бір де бір төбе қосыла алмайды. Демек, {d,e,f} тығыз байланысқан ішкі жиын болып табылады. Графтың {d,e,f} жиынынан алынған тығыз байланысқан компоненті 22-суретте келтірілген. Егер граф тығыз байланысқан граф болмаса, онда оның гамильтон контуры болмайды. Бұл мынадан шығады: гамильтон контуры граф төбелерінің әрбір жұбының арасындағы жолды қамтиды. Осылайша, гамильтон контурының бар болуының қажетті шарты гарфта тығыз байланыстың болуы болып табылады. 7-тарауда айтылғандай, ілмек деп бастапқы және соңғы төбелері сәйкес келетін, яғни (x,x) түріндегі доғаларды айтамыз. Гамильтон контурында ілмек болуы мүмкін емес, және демек, G графындағы гамильтон контурының бар болуына ілмектерді қосу немесе алып тастау әсер етпейді. Егер екі төбе, мысалы x және y, бағыттары бірдей $(x,y)_1$, $(x,y)_2$,… бірнеше доғамен байланысқан болса, онда ең қысқа ұзындығы бар доғадан басқа барлық доғалардың алып тасталуы гамильтон контурының болуына да, графтың оңтайлы гамильтон контурының ұзындығына да әсер етпейді, егер ол бар болса.

Осы себепті G графында ілемектер жоқ деп алынады және кез келген x пен y үшін x-тан y-ке апаратын доғалардың саны 1-ден аспайды деп қабылданады.

Коммивояжер есебін шешудің көптеген әр түрлі әдістері бар, олардың авторлары – Белмор мен Немхаузер, Гарфипкель мен Немхаузер, Хелд пен Карп, Стекхан [9]. Осы әдістердің барлығы келесі екі кластың біріне жатады:

1. Әрқашан да оңтайлы шешімнің табылуына алып келетін, бірақ ол үшін өте көп операцияны талап ететін әдістер;

2. Оңтайлы шешімнің табылуына үнемі алып келе бермейтін, бірақ талап етілетін операциялар саны қолайлы болып табылатын әдістер.

Коммивояжер есебін шешудің көптеген танымал әдістерінің негізінде сызықтық және бүтін сандық

бағдарламалау теориясынан алынған нәтижелер жатыр. Осы әдістердің біріншісі – бұтақтар мен шекаралар әдісі – ол 1-класқа жатады, екіншісі – шешімді біртіндеп жақсарту әдісі – ол 2-класқа жатады. Берілген оқулықта коммивояжер есебі тағайындаулар туралы есеп ретінде және динамикалық бағдарламалау есебі ретінде қарастырылады.

8.1. Коммивояжер есебі тағайындаулар туралы есептің бір түрі ретінде

Коммивояжер туралы есептің математикалық моделі:

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

$$x_{ij} = \begin{cases} 1, \text{ егер коммивояжер i-ші қаладан кейін j-ші қалаға барса;} \\ 0, \text{ кері жағдайда} \end{cases}$$

Коммивояжер жолды минималдай отырып, бір реттен кіріп шығуы тиіс n пункт бар және ол бастапқы пунктке қайтып оралуы керек.

Негізгі ұғымдар:

$C=\|c_{ij}\|$ – c_{ij} элементтері i-ші қаладан j-ші қалаға өту кезіндегі шығындарды анықтайтын матрица болсын. (i, j) қалалары жұпты құрайды.

t циклы деп реттелген n қалалар жұбының жиынын атаймыз, олар әрбір қаладан тек бір реттен ғана өтетін маршрутты құрайды.

$t=[(i_1, i_2), (i_2, i_3),\ldots, (i_{n-1}, i_n), (i_n, i_1)]$.

Әрбір (i, j) жұбы маршрут коммуникациясын құрайды. Сонда t циклы үшін C матрицасына сәйкес келетін z(t) жалпы шығындары – бұл маршрут коммуникациялары бойынша матрица элементтерінің қосындысы: $z(t) = \sum_{(c_{ij}) \in t} c_{ij}$.

Циклда әрбір жолдың және әрбір бағанның тек бір элементі ғана болады.

Егер C матрицасының қандай да бір і-ші жолының немесе j-ші бағанының бүкіл элементтерінен олардың минималдысын алып тастайтын болсақ, әрбір жолында және әрбір бағанында ең болмаса бір нөлі бар матрицаны аламыз. Алынған матрица келтірілген матрица деп аталады, ал нөлдерді түзу процесі – келтіру деп аталады.

Келтіру процесіндегі азайтылатын элементтердің қосындысы келтіруші константа деп аталады және $h^{(k)}$ деп белгіленеді, мұндағы k – келтірудің реттік нөмірі.

$C_{i,j(i)}=\min_j C_{ij}$ болсын, онда $C'_{ij}=C_{ij}-C_{i,j(i)}$ (8.2)

$C^*_{i(j),j}=\min_i C'_{ij}$ болсын, онда $C''_{ij}=C'_{ij}-C^*_{i(j),j}$ (8.3)

$$h^{(1)} = \sum_{i=1}^{n} C_{i,j(i)} + \sum_{j=1}^{n} C^*_{i(j),j} .$$

Егер Z(t) – келтіруге дейінгі матрица үшін t циклының шығындары болса,

$Z^1(t)$ – келтіруден кейінгі матрица үшін t циклының шығындары болса,

$h^{(1)}$ – келтіруші константалардың қосындысы болса, онда $Z(t)=h^{(1)}+Z^1(t)$.

Келтірілген матрицада тек қана теріс элементтер болғандықтан, онда $h^{(1)}$ ескі матрицадағы t циклының шығындарының төменгі шекарасы болып табылады.

Литтл алгоритмі:

1^0. k=1.

2^0. C матрицасын жолдары мен бағандары бойынша келтіруді орындап, келтірілген C^k матрицасын аламыз.

3^0. $h^{(k)}$ келтіруші константаларының қосындысын есептейміз – бұл X жиыны үшін баға, оны $w(x)$ деп белгілейік.

4^0. Y жиынына қосу үшін үміткерлерді таңдаймыз, яғни $C^k_{ij}=0$ болатындай барлық $(i,j), i,j = \overline{1,n},\ i \neq j.$

5^0. Y жиынына қосуға белгіленген үміткерлер үшін есептейміз:

$$\theta(i,j) = \min_{j' \neq j} C_{ij'} + \min_{i' \neq i} C_{i'j}.$$

6^0. $C^k_{ij}=0$ болатындай $\theta(k,l) = \max \theta(i,j), \forall i,j$ таңдаймыз. (k, l) жұбы Y жиынына кіреді және $\overline{Y}$ жиынына кіре алмайды.

7^0. $\bar{Y}$ жиыны үшін бағасын есептейміз, ол X жиынына жұмсалған шығындарға және $\theta(k,l)$-ге тең, яғни

$$w(\bar{Y}) = w(x) + \theta(k,l).$$

8^0. Әрбір қаладан (пункттен) тек бір рет қана шығуға болатындықтан, k-шы жолды ары қарай қарастырмау үшін сызып тастаймыз, әрбір қалаға тек бір рет қана кіруге болатындықтан, l-ші бағанды да сызып тастаймыз. Тұйықталған ішкі циклдарды болдырмау үшін l-ден k-ға баратын жолға тыйым салу керек, яғни (l, k) ұяшығына X немесе ∞ қоямыз.

9^0. Пайда болған қысқартылған матрица қандай да бір бұтақталу қадамында 2x2 өлшеміне келеді де, оның ішінде тек екі қала жұбы болады. Бұл жұптар ілмектері жоқ қандай да бір маршрут үшін тұйықтаушы жұптар болады. Сонымен, 2x2 матрицасының түзілу сәті ерекше болып табылады, сондықтан 9^0 қадамда тексереміз: пайда болған матрицамыздың өлшемі 2x2 ме? Егер иә болса, онда 11^0 қадамға көшеміз, жоқ болса, 10^0 қадамға көшеміз.

10^0. Алынған матрица келтірілген матрица ма? Егер иә болса, онда Y жиынының бағасы Y алынған жиынның бағасына тең болады, яғни $w(Y) = w(X)$. Егер жоқ болса, онда жаңа ғана алынған матрицаның келтірілуі жүзеге асырылады да, $h^{(k+1)}$ есептеледі, содан кейін:

$w(Y) = w(X) + h^{(k+1)}, k = k+1,$ сосын 4^0 қадамға көшеміз.

11^0. Оңтайлылық шартын тексеру: егер тұйықталған циклдың бағасы ары қарай бұтақталатын барлық жиындардың бағасынан жоғары болмаса, онда алынған тұйықталған маршрут оңтайлы маршрут болып табылады. Егер бағасы төмен болатын ең болмаса бір маршрут бар болса, онда алынған тұйықталған маршрут есте сақталады. Содан кейін бұтақталу процесі төмен бағасы бар жиынмен жалғастырылады

17-мысал.

Коммивояжер бір реттен кіріп шығуы тиіс бес пункт бар, және ол бастапқы пунктке қайтып келуі тиіс. Коммивояжердің ең қысқа жолын анықтау керек. Жол шығындары $C=\|c_{ij}\|$ матрицасымен берілген.

102-кесте

	1	**2**	**3**	**4**	**5**
1	x	4	10	9	4
2	2	x	9	7	6
3	8	5	x	5	9
4	5	8	10	x	7
5	1	5	4	8	x

Матрицаны жол бойынша келтірейік. Жолдар бойынша минималды элементтер: (4,2,5,5,1), оларды сәйкес жолдардың әрбір элементінен азайтамыз:

103-кесте

	1	**2**	**3**	**4**	**5**
1	x	0	6	5	0
2	0	x	7	5	4
3	3	0	x	0	4
4	0	3	5	x	2
5	0	4	3	7	x

Матрицаны баған бойынша келтірейік. Әрбір баған бойынша минималды элементтер: (0,0,3,0,0), оларды сәйкес бағандардың әрбір элементінен азайтамыз:

104-кесте

	1	**2**	**3**	**4**	**5**
1	x	0	3	5	0
2	0	x	4	5	4
3	3	0	x	0	4
4	0	3	2	x	2
5	0	4	0	7	x

Келтіруші константа $h^{(1)}=20$.

Бұтақталуға үміткерлерді таңдаймыз да, олардың бағасын есептейміз:

$C_{12}=0$, $\Theta(1,2)=0$
$C_{15}=0$, $\Theta(1,5)=2$
$C_{21}=0$, $\Theta(2,1)=4$
$C_{32}=0$, $\Theta(3,2)=0$
$C_{34}=0$, $\Theta(3,4)=5$
$C_{41}=0$, $\Theta(4,1)=2$
$C_{51}=0$, $\Theta(5,1)=0$
$C_{53}=0$, $\Theta(5,3)=2$

Барлық бағалардың арасынан ең үлкенін таңдаймыз, ол 5-ке тең және (3,4) жұбы Y жиынына қосылады. 3-ші жол мен 4-ші бағанды матрицадан сызып тастаймыз. (4,3) элементінің мәні (x)-ке айналады. Y жиынының бағасы 25-ке тең. Осы түрлендірулерден кейін келесі матрицаны аламыз:

105-кесте

	1	**2**	**3**	**5**
1	x	0	3	0
2	0	x	4	4
4	0	3	x	2
5	0	4	0	x

Матрицаны жолдары мен бағандары бойынша келтіреміз.

Келтіруші константа $h^{(2)}=0$.

Бұтақталуға үміткерлерді таңдаймыз да, олардың бағасын есептейміз:

$C_{12}=0$, $\Theta(1,2)=3$
$C_{15}=0$, $\Theta(1,5)=2$
$C_{21}=0$, $\Theta(2,1)=4$
$C_{41}=0$, $\Theta(4,1)=2$
$C_{51}=0$, $\Theta(5,1)=0$
$C_{53}=0$, $\Theta(5,3)=2$

Барлық бағалардың арасынан ең үлкенін таңдаймыз, ол 4-ке тең және (2,1) жұбы Y жиынына қосылады. 2-ші жол мен 1-ші бағанды матрицадан сызып тастаймыз. (1,2) элементінің мәні (х)-ке айналады. Y жиынының бағасы 24-ке тең. Осы түрлендірулерден кейін келесі матрицаны аламыз:

106-кесте

	2	**3**	**5**
1	x	3	0
4	3	x	2
5	4	0	x

Матрицаны жолдары бойынша келтіреміз. Жолдар бойынша минималды элементтер: (0,2,0). Оларды сәйкес жолдың әрбір элементінен азайтамыз:

107-кесте

	2	**3**	**5**
1	x	3	0
4	1	x	0
5	4	0	x

Матрицаны бағандары бойынша келтіреміз. Әрбір баған бойынша минималды элементтер: (1,0,0), оларды сәйкес бағанның әрбір элементінен азайтамыз. Келесі матрицаны аламыз:

108-кесте

	2	**3**	**5**
1	x	3	0
4	0	x	0
5	3	0	x

Келтіруші константа $h^{(3)}=3$.

Бұтақталуға үміткерлерді таңдаймыз да, олардың бағасын есептейміз:

$C_{15}=0$, $\Theta(1,5)=3$

$C_{42}=0$, $\Theta(4,2)=0$

$C_{45}=0$, $\Theta(4,5)=0$

$C_{53}=0$, $\Theta(5,3)=6$

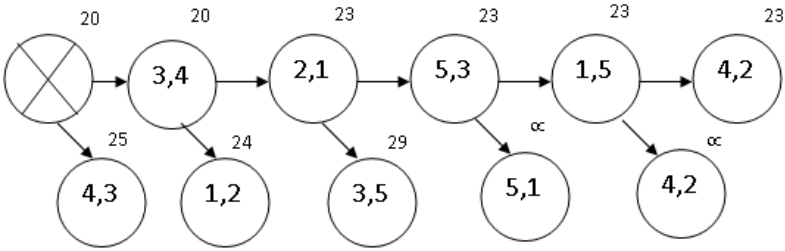
Барлық бағалардың арасынан ең үлкенін таңдаймыз, ол 6-ға тең және (5,3) жұбы Y жиынына қосылады. 5-ші жол мен 3-ші бағанды матрицадан сызып тастаймыз. Y жиынының бағасы 29-ға тең. Осы түрлендірулерден кейін келесі матрицаны аламыз:

109-кесте

	2	**5**
1	x	0
4	0	0

Матрицамыз келтірілген матрица болып шықты, сондықтан келтіруші константа $h^{(4)}=0$.

Соңғы 2x2 өлшемді матрицадан Y жиынына екі жұпты таңдаймыз: (1,5) жұбы және (4,2) жұбы. Бұтақты құрайық:



23-сурет

Шешімі:

$$3 \to 4 \to 2 \to 1 \to 5 \to 3$$

$$z^* = 23$$

8.2. Р. Беллман алгоритмін коммивояжер есебіне қолдану

n+1 қала бар, олардың ара қашықтықтары белгілі, олар $C = \| c_{ij} \|$ матрицасымен берілген. Жалпы жол минимал болатындай етіп, қалаларды аралап шығудың тұйықталған циклын құру керек.

Кейбір белгілеулерді енгізейік. Коммивояжер шыққан қаланы нөлінші қала деп атайық. Ізделініп отырған маршрутымыз тұйық болғандықтан, берілген жиындағы кез келген қаланы нөлінші қала деп атауға болады. $i_1, i_2, \ldots, i_n$ маршрутымен өткен жолдың $c_{0i1} + c_{i1i2} + \ldots + c_{in0}$ ұзындығы минимал болатындай етіп, 1, 2, …, n нөмірлі қалалардың орын ауыстыруын $\pi = (i_1, i_2, \ldots, i_n)$ деп белгілейік. Еркін ретпен $i_1, i_2, \ldots, i_{k-1}$ қалалар арқылы өтетін 0 және i_k қалаларының арасындағы ең қысқа жолдың ұзындығын $f_{k-1}(0; i_1, i_2, \ldots, i_{k-1}, i_k)$ деп, ал оған сәйкес орын ауыстыруды $\pi_{k-1}(0, i_1, i_2, \ldots, i_{k-1}, i_k)$ деп белгілейік. Оңтайлы тізбекті құру алгоритмі келесі жайтқа негізделеді:

$(i'_1, i'_2, \ldots, i'_{k-1}) = \pi_{k-1}(0; i_1, i_2, \ldots, i_{k-1}, i_k)$, $1 \le r \le k-1$ болсын, сонда $(i'_1, i'_2, \ldots, i'_r) = \pi_r(0; i_1, i_2, \ldots, i_r, i_{r+1})$,

Яғни оңтайлы тізбек оның кез келген тізбегі үшін оңтайлы болып табылады.

π_n (0, i_1, i_2, …, i_{n-1}, i_n, 0) құру алгоритмін жазайық:

0. Келесі функцияларды есептейміз: f_0 (0; i) = c_{0i}, i = 1, 2, …, n. Есептеулер саны P_0=n. Әрбір f_0 (0; i) есте сақталады.

1. f_1 (0; i, j) есептейміз:

f_1 (0; i, j) = f_0 (0; i) + c_{ij}. i, j = 1, 2, …, n, i≠j.

Динамикалық бағдарламалау әдісінің жалпы сұлбасына сәйкес, f_1 (0; i, j) есте сақталады, f_0 (0; i) өшіріледі.

Есептелуі тиіс f_1 (0; i, j) мәндерінің саны, $P_1 = nC^1_{n-1} = n(n-1)$; осы қадамда табылған мәндерді есте сақтау үшін осындай жады көлемі қажет.

k. $2 \le k \le n-1$. Есептейміз

f_k (0; i_1, i_2, …, i_k, i_{k+1}) = min {[f_{k-1} (0; i_2, …, i_k, i_1)+c_{i1}, i_{k+1}],

[f_{k-1} (0; i_1, i_3, …, i_k, i_2) + c_{i2ik+1}], … [f_{k-1} (0; i_1, i_2, …, i_{k-1}, i_k) + c_{ikik+1}]},

Әрбір f_k (0; i_1, i_2, …, i_k, i_{k+1}) мәні үшін π_k (0; i_1, i_2, …, i_k, i_{k+1}) оңтайлы орын ауыстыруы есте сақталады. Бұл қадамда есептелген мәндердің саны $P_k = nC^k_{n-1}$, оларды есептеуге қажетті жады көлемі f_k-ны есептеуге қажетті f_{k-1}, π_{k-1} мәндерінің санымен анықталады.

Ол $nC^{k-1}_{n-1} + nC^k_{n-1}$ құрайды.

n. Есептейміз

f_n(0; 1, 2, …, n, 0) = min {[f_{n-1} (0; 2, …, n, 1)+c_1,0], [f_{n-1} (0; 1, 3, …, n, 2) + $c_{2,0}$], … [f_{n-1} (0; 1, 2, 3, …, n-1, n) + $c_{n,0}$]}.

Минимумға қол жеткізілетін π_n (0; 1, 2, …, n, 0) орын ауыстыруы бастапқы есеп үшін оңтайлы орын ауыстыру болып табылады.

Есептелуі тиіс f_n мәндерінің саны P_n = 1, егер жалғыз оңтайлы цикл бар болса. Егер C матрицасы симметриялы болса, онда қалаларды аралап шығудың қарама-қарсы ретімен бірдей ұзындықтағы екі цикл бар болады.

f_n, π_n есептеуге қажетті жады көлемі $nC^{n-1}_{n-1} + 1 = n+1$.

Алдыңғы парагафтағы есепті Беллман әдісімен шешейік.

1-ші итерация.

f_0(1;2)=4,

f_0(1;3)=10,

f_0(1;4)=9,

f_0(1;5)=4.

2-ші итерация.

$f_1(1;2,i)= f_0(1;2)+c_{23}=4+9=13,$
$f_1(1;2,i)= f_0(1;2)+c_{24}=4+7=11,$
$f_1(1;2,i)= f_0(1;2)+c_{25}=4+6=10.$

$f_1(1;3,i)= f_0(1;3)+c_{32}=10+5=15,$
$f_1(1;3,i)= f_0(1;3)+c_{34}=10+5=15,$
$f_1(1;3,i)= f_0(1;3)+c_{35}=10+9=19.$

$f_1(1;4,i)= f_0(1;4)+c_{45}=9+7=16,$
$f_1(1;4,i)= f_0(1;4)+c_{43}=9+10=19,$
$f_1(1;4,i)= f_0(1;4)+c_{42}=9+8=17.$

$f_1(1;5,i)= f_0(1;5)+c_{54}=4+8=12,$
$f_1(1;5,i)= f_0(1;5)+c_{53}=4+4=8,$
$f_1(1;5,i)= f_0(1;5)+c_{52}=4+5=9.$

3-ші итерация.

$f_2(1;2,i,j)= \min\{ f_1(1;2,3) + c_{34}; f_1(1;2,3) + c_{35}; f_1(1;2,4) + c_{43}; f_1(1;2,4) + c_{45}; f_1(1;2,5) + c_{53}; f_1(1;2,5) + c_{54}\}=\min\{13+5; 13+9; 11+10; 11+7; 10+4; 10+8\}=\min\{18; 22; 21; 18; 14; 18\}=14;$

$\pi^*(1,2,5,3).$

$f_2(1;3,i,j)= \min\{ f_1(1;3,2) + c_{24}; f_1(1;3,2) + c_{25}; f_1(1;3,4) + c_{45}; f_1(1;3,4) + c_{42}; f_1(1;3,5) + c_{54}; f_1(1;3,5) + c_{52}\}=\min\{15+7; 15+6; 15+7; 15+8; 19+8; 19+5\}=\min\{22; 21; 22; 23; 27; 24\}=21;$

$\pi^*(1,3,2,5).$

$f_2(1;4,i,j)= \min\{ f_1(1;4,5) + c_{52}; f_1(1;4,5) + c_{53}; f_1(1;4,3) + c_{35}; f_1(1;4,3) + c_{32}; f_1(1;4,2) + c_{23}; f_1(1;4,2) + c_{25}\}=\min\{16+5; 16+4; 19+9; 19+5; 17+9; 17+6\}=\min\{21; 20; 28; 24; 26; 23\}=20;$

$\pi^*(1,4,5,3).$

$f_2(1;5,i,j)= \min\{ f_1(1;5,4) + c_{42}; f_1(1;5,4) + c_{43}; f_1(1;5,3) + c_{34}; f_1(1;5,3) + c_{32}; f_1(1;5,2) + c_{23}; f_1(1;5,2) + c_{24}\}=\min\{12+8; 12+10; 8+5; 8+5; 9+9; 9+7\}=\min\{20; 22; 13; 13; 18; 16\}=13;$

$\pi^*(1,5,3,4)$ и $\pi^*(1,5,3,2).$

4-ші итерация.

$f_3(1;i,j,k)= \min\{ f_2(1;2,5,3) + c_{34}; f_2(1;3,2,5) + c_{54}; f_2(1;4,5,3) + c_{32}; f_2(1;5,3,4) + c_{42}; f_2(1;5,3,2) + c_{24}\}=\min\{14+5; 21+8; 20+5; 13+8; 13+7\}=\min\{19; 29; 25; 21; 20\}=19;$

$\pi^*(1,2,5,3,4)$.

5-ші итерация.

$f_4(1;i,j,k,1)= \min\{ f_3(1;2,5,3,4) + c_{41}; f_3(1;3,2,5,4) + c_{41}; f_3(1;4,5,3,2) + c_{21}; f_3(1;5,3,4,2) + c_{21}; f_3(1;5,3,2,4) + c_{41}\}=\min\{19+5; 29+5; 25+2; 21+2; 20+5\}=\min\{24; 34; 27; 23; 25\}=23;$

$\pi^*(1,5,3,4,2,1)$.

Жауабы: Функцияның минимумына (f=23) қол жеткізілетін $\pi^*(1,5,3,4,2,1)$ орын ауыстыруы бастапқы есептің оңтайлы орын ауыстыруы болып табылады.

Сұрақтар мен тапсырмалар

1. Келтірілген матрицаны алу ережелерін айтыңыз.
2. Келтіруші константаны анықтау ережелері.
3. Бұтақталуға үміткерлерді анықтау ережелерін тұжырымдаңыз.
4. Коммивояжер туралы есептің қойылымын келтіріңіз.
5. Матрицаның нөлдерін қалыптастыру процесі қалай аталады?
6. Коммивояжер есебі есептердің қандай класына жатады?
7. Бұтақталу үшін белгіленген үміткердің бағасын есептеу формуласын келтіріңіз.
8. Қандай жағдайларда жалғыз оңтайлы цикл бар болады?
9. Маршрут коммуникациясы деген не?
10. Келесі есепті Литтл алгоритмімен шығарыңыз:

	1	**2**	**3**	**4**	**5**	**6**
1	×	39	45	2	51	33
2	30	×	20	33	40	35
3	54	16	×	55	22	56
4	19	36	25	×	18	43
5	29	8	8	12	×	25
6	16	47	31	14	8	×

11. Келесі есепті Литтл алгоритмімен шығарыңыз:

	1	**2**	**3**	**4**	**5**	**6**
1	×	1	7	3	14	2
2	3	×	6	9	1	24
3	6	14	×	3	7	3
4	2	3	5	×	9	4
5	15	7	11	2	×	4
6	20	5	13	4	13	×

12. Келесі есепті Литтл алгоритмімен шығарыңыз:

	1	**2**	**3**	**4**	**5**	**6**
1	×	15	43	38	10	45
2	44	×	18	6	49	40
3	41	42	×	19	1	48
4	33	44	20	×	20	21
5	40	17	16	26	×	15
6	3	4	37	54	36	×

13. Келесі есепті Литтл алгоритмімен шығарыңыз:

	1	**2**	**3**	**4**	**5**	**6**
1	×	28	29	2	2	58
2	48	×	41	13	51	23
3	56	53	×	59	46	57
4	5	31	48	×	53	20
5	26	54	42	48	×	27
6	43	22	15	30	9	×

14. Келесі есепті Беллман алгоритмімен шығарыңыз:

	1	**2**	**3**	**4**	**5**	**6**
1	×	15	43	38	10	45
2	44	×	18	6	49	40
3	41	42	×	19	1	48
4	33	44	20	×	20	21
5	40	17	16	26	×	15
6	3	4	37	54	36	×

15. Келесі есепті Беллман алгоритмімен шығарыңыз:

	1	**2**	**3**	**4**	**5**	**6**
1	×	19	40	30	15	9
2	56	×	14	57	12	8
3	42	4	×	28	33	24
4	1	12	5	×	21	16
5	21	16	38	52	×	59
6	51	1	17	22	29	×

16. Келесі есепті Беллман алгоритмімен шығарыңыз:

	1	**2**	**3**	**4**	**5**	**6**
1	×	10	7	36	40	36
2	34	×	58	48	33	53
3	47	30	×	59	10	15
4	31	12	15	×	22	11
5	9	16	3	13	×	9
6	36	4	14	20	48	×

9. БЕЙСЫЗЫҚТЫҚ БАҒДАРЛАМАЛАУ

9.1. Негізгі ұғымдар. Бейсызықтық бағдарламалау әдістерінің жіктелуі

Көптеген инженерлік есептерде математикалық модельдердің құрылуын сызықтық бағдарламалау есептеріне келтіру мүмкін болмай жатады.

Шынайы объектілерді немесе технологиялық процестерді жобалау есептерінің математикалық модельдері олардың ішінде өтіп жатқан шынайы физикалық және бейсызықтық процестерді бейнелеуі тиіс. Осы объектілер мен процестердің айнымалылары өзара физикалық бейсызықтық заңдармен, мысалы, массаның немесе энергияның сақталу заңы сияқты заңдармен байланысқан. Нәтижесінде, ғылыми-зерттеу жобаларында және жобалау есептерінде кездесетін математикалық бағдарламалау есептерінің көпшілігі – бұл бейсызықтық бағдарламалау есептері (ББЕ).

Ең ауқымды мағынасында бейсызықтық бағдарламалаудың жалпы есебі теңдіктер және (немесе) теңсіздіктер түріндегі шектеулері берілген жағдайда мақсатты функцияның экстремумын іздеп табу болып табылады. Бейсызықтық бағдарламалау есептерінің мақсатты функциясы бейсызықтық болады, ал шектеулері сызықтық немесе бейсызықтық болуы мүмкін.

Бейсызықтық бағдарламалау есебінің жалпы түрі мынадай:

$$f(x_1, x_2, \dots, x_n) \to \min(\max) \tag{9.1}$$

шарттары:

$$x \in S,$$
$$S = \left\{x \middle| g_i(x_1, x_2, \dots, x_n) \le 0,\ i = \overline{1,l},\ h_i(x_1, x_2, \dots, x_n) = 0,\ i = \overline{1,k}\right\}. \tag{9.2}$$

мұндағы f(x), $g_i(x)$ және $h_i(x)$ – п айнымалыларының кейбір белгілі функциялары.

(9.2) қатынастары шектеулер жүйесін құрайды және айнымалылардың теріс еместік шартын қамтиды, егер ондай шарттар бар болса. Айнымалылардың теріс еместік шартын тікелей беруге де болады.

E_n эвклид кеңістігінде (9.2) шектеулер жүйесі есептің мүмкін шешімдерінің облысын анықтайды. Сызықтық бағдарламалау есебімен салыстырғанда, ол әрқашан да дөңес бола бермейді.

Егер мүмкін шешімдер облысы анықталған болса, онда (9.1)-(9.2) есебінің шешілуі осы облыстың ең жоғарғы деңгейлі гипер кеңістік өтетін нүктесін табуға келтіріледі: $\mathbf{f(x_1, x_2, \dots, x_n)}$. Көрсетілген нүкте мүмкін шешімдер облысының шекарасында да, оның ішінде де бола алады.

Бейсызықтық бағдарламалау есебін шешу кезіндегі маңызды ұғым – градиент ұғымы. Градиент деп f(x)-тің x бойынша бірінші дербес туындыларынан құралатын вектор-бағанды айтамыз:

$$\nabla f(x) = \begin{pmatrix} \frac{\partial f(x)}{\partial x_1} \\ \frac{\partial f(x)}{\partial x_n} \end{pmatrix}$$

мұндағы n – f(x) функциясының евклид кеңістігінің өлшемінің саны, яғни f(x) функциясына кіретін айнымалылар саны.

Градиенттің маңыздылығы мынада: функцияның әрбір нүктесінде ол функцияның барынша тез арту жағына қарай бағытталған, яғни функцияның максимумын табу үшін қай жаққа қарай жылжу керектігін көрсетеді. Сәйкесінше, теріс таңбамен алынған градиент функцияның барынша тез құлдилау (азаю) бағытын көрсетеді. Экстремум нүктесінде градиент мәні нөлге тең.

Бүгінгі таңда бейсызықтық бағдарламалау есебін шешу үшін әр түрлі күрделілік деңгейіндегі көптеген әдістер

ұсынылған. Бұл әдістердің ешқайсысы әмбебап болып табылмайды және бейсызықтық есептердің барлық түріне келе бермейді. Әрбір әдістің өзіндік артықшылықтары мен кемшіліктері бар. Әзірше кез келген бейсызықтық бағдарламалау есебін шешуге арналған әдіс жасалмады, және қазіргі қолданыстағы әдістердің арасынан таңдаудың бірыңғай әдістемесі ұсынылмады. Сондықтан да, есепті шешу әдісін таңдау нақты жағдайға, есептің түріне, мақсатты функцияның қасиеттеріне байланысты. Сонда да бейсызықтық бағдарламалау есебін шешу әдістерін келесідей белгілері бойынша жіктеуге болады:

Мақсатты функциядағы локальды критерийлердің санына байланысты бейсызықтық бағдарламалау әдістері былайша бөлінеді:

- бір критерийлі,
- көп критерийлі.

$\overline{X}$ векторының ұзындығына байланысты әдістер былайша бөлінеді:

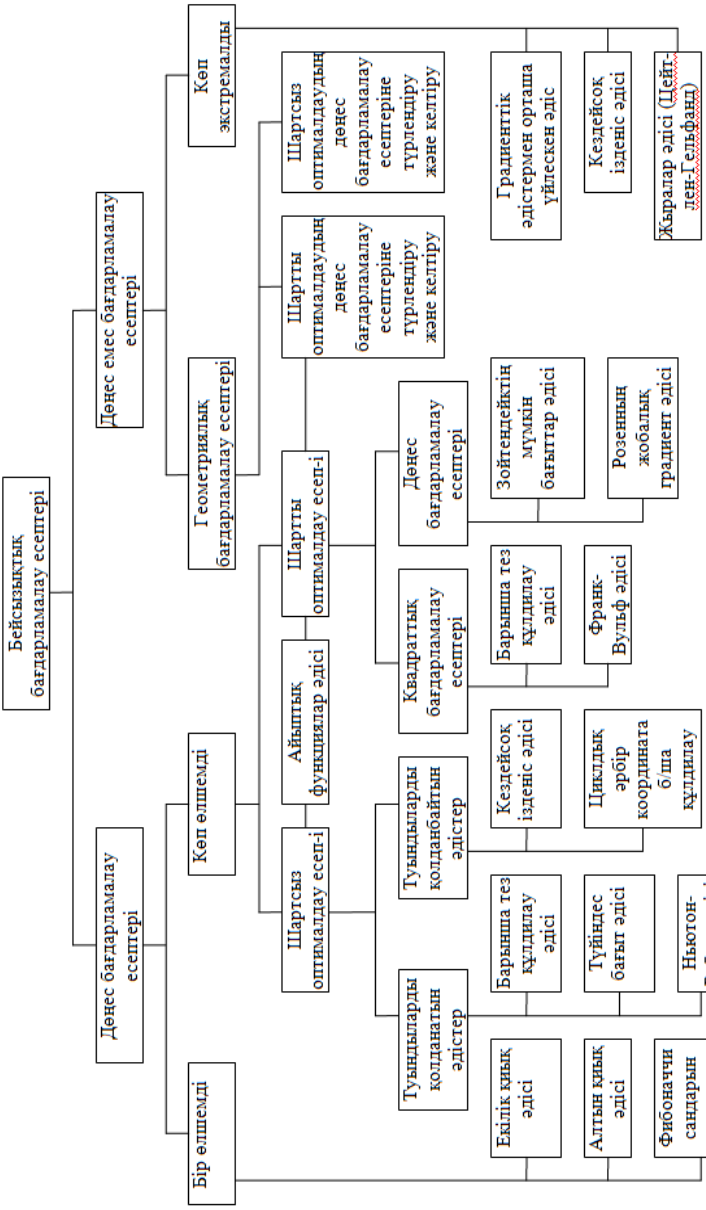
- бір параметрлі немесе бір өлшемді n=1),
- көп параметрлі немесе көп өлшемді (n>1).

Шектеулердің болуына байланысты бейсызықтық бағдарламалау есептері былайша бөлінеді:

- шектеулерсіз (шартсыз оңтайландыру),
- шектеулермен (шартты оңтайландыру).

Экстремумды іздеу алгоритмінде қолданылатын ақпарат типіне байланысты әдістер былайша бөлінеді:

- тікелей іздеу әдістері, яғни мақсатты функцияның экстремумын іздеу кезінде оның мәндері ғана қолданылатын әдістер;
- бірінші ретті градиенттік әдістер, мұнда функцияның экстремумын іздеу кезінде олардың бірінші туындыларының мәндері қолданылады;
- екінші ретті градиенттік әдістер, мұнда функцияның экстремумын іздеу кезінде олардың бірінші туындыларымен бірге екінші туындылары да қолданылады.



24-сурет – Бейсызықтық оптималдау есептерін шешу әдістерінің жіктелуі

Сондықтан әрбір нақты жағдайда қолданылатын әдісті шешіліп отырған есептің ерекшеліктеріне бейімдеу керек. Бейсызықтық бағдарламалау есебін шешу әдістерінің жіктелуі 24-суретте келтірілген.

9.2 Бейсызықтық бағдарламалау есептерінің қос мағыналылығы

Тіке есеп	Қос мағыналы есеп
$f(x_1, x_2, \dots, x_n) \le b_i,\ i = \overline{1,l},$	$L(x,u,v) = f(x) + \sum_{i=1}^{l} g_i(x) + \sum_{i=1}^{k} v_i h_i(x)$
$\{\lvert g_i(x_1, x_2, \dots, x_n) \le b_i,\ i = \overline{1,l},\}$	$u_i \ge 0$
$\{\lvert h_i(x_1, x_2, \dots, x_n) \le b_i,\ i = \overline{1,k}\}$	$v_i >< 0$
$x_j \ge 0$	$\frac{\partial f}{\partial x_j} + \sum_{i=1}^{l} u_i \frac{\partial g_i}{\partial x_j} + \sum_{i=1}^{k} v_i \frac{\partial h_i}{\partial x_j} \ge 0,$ $j = \overline{1,k}$
x_j	$\frac{\partial f}{\partial x_j} + \sum_{i=1}^{l} u_i \frac{\partial g_i}{\partial x_j} + \sum_{i=1}^{k} v_i \frac{\partial h_i}{\partial x_j} = 0,$ $j = \overline{k+1,n}$

Қос мағыналы есепті алу ережелері:

1. Қос мағыналы есептің мақсатты функциясы келесі түрдегі жалпыланған Лагранж функциясы болып табылады:

$$L(x,u,v) = f(x) + \sum_{i=1}^{l} g_i(x) + \sum_{i=1}^{k} v_i h_i(x)$$

2. Тіке есептің мақсатты функциясының минимальдануы қос мағыналы есептің мақсатты функциясының максимальдануымен алмастырылады.

3. Тіке есептің әрбір ($g_i(x) \le b_i$) теңсіздік-шектеуіне қос мағыналы есептің $u_i \ge 0$ еркін емес айнымалысы сәйкес келеді.

4. Тіке есептің әрбір ($h_i(x) = b_i$) теңдік-шектеуіне қос мағыналы есептің $v_i \gtrless 0$ еркін айнымалысы сәйкес келеді.

5. Тіке есептің әрбір $x_j \geq 0$ еркін емес айнымалысына қос мағыналы есептің теңсіздік-шектеуі сәйкес келеді:

$$\frac{\partial f}{\partial x_j} + \sum_{i=1}^{l} u_i \frac{\partial g_i}{\partial x_j} + \sum_{i=1}^{k} v_i \frac{\partial h_i}{\partial x_j} \geq 0, j = \overline{1,k}$$

6. Тіке есептің әрбір еркін айнымалысына қос мағыналы есептің теңдік-шектеуі сәйкес келеді:

$$\frac{\partial f}{\partial x_j} + \sum_{i=1}^{l} u_i \frac{\partial g_i}{\partial x_j} + \sum_{i=1}^{k} v_i \frac{\partial h_i}{\partial x_j} = 0,\ j = \overline{k+1,n}.$$

18-мысал.

Тіке есеп	Қос мағыналы есеп
$z = x_1^2 + x_2^2 + x_3^2 - 2x_1 + x_2 \to \min$	$L(x,u,\upsilon) = x_1^2 + x_2^2 + x_3^2 - 2x_1 + x_2 +$ $+ U_1(x_1 + 2x_2 + 3x_3 - 13) + U_2(2x_1 + x_2^2 +$
$x_1 + 2x_2 + 3x_3 \leq 13$	$x_3 - 6) + \upsilon(x_1^2 + 2x_2 - 8) \to \max$
$2x_1 + x_2^2 + x_3 \leq 6$	$u_1 \geq 0$
$x_1^2 + 2x_2 = 8$	$u_2 \geq 0$
$x_1 \geq 0$	υ
x_2	$2x_1 - 2 + u_1 + 2u_2 + 2\upsilon x_1 \geq 0$
$x_3 \geq 0$	$2x_2 + 1 + 2u_1 + 2u_2 x_2 + 2\upsilon = 0$
	$2x_3 + 3u_1 + u_2 \geq 0$

9.3. Қос мағыналылық теоремалары

Теорема 6. (Сызықтық бағдарламалаудың бірінші қос мағыналылық теоремасын қорыту). Егер $f(x)$, $g_i(x)$ және $h_i(x)$ функциялары дифференциалданатын және дөңес функциялар болса, және x^* тіке есептің шешімі болса, онда қос мағыналы есептің шешімі болып табылатын (x^*, u^*, v^*) векторы табылады және екі есептің де экстремумдары сәйкес келеді.

Теорема 7. (сызықтық бағдарламалаудың екінші қос мағыналылық теоремасын қорыту). $f(x)$, $g_i(x)$ және $h_i(x)$ функциялары екі рет дифференциалданатын болсын, және $f(x)$ функциясы немесе $g_i(x^*) = 0$ шартын қанағаттандыратын $g_i(x)$ функцияларының бірі қатаң дөңес болсын, сондай-ақ (x^*, u^*, v^*) векторы қос мағыналы есептің шешімі болсын, онда x^* қос мағыналы есептің шешімі болады, және x^* және (x^*, u^*, v^*) оңтайлы шешімдері үшін $u_i^* g_i(x^*) = 0$ қатынасы орындалады.

9.4. Жалпы бейсызықтық бағдарламалау есебінің дербес жағдайы

(9.1)-(9.2) бейсызықтық бағдарламалау есебінің дербес жағдайын қарастырайық. Ол үшін шектеулер жүйесінде тек теңдіктер ғана бар деп, айнымалылардың теріс еместік шарты жоқ деп, және $f(x_1,x_2,\ldots,x_n)$ және $h(x_1,x_2,\ldots,x_n)$ – өздерінің дербес туындыларымен бірге үздіксіз функциялар деп алайық

$$f(x_1, x_2, \ldots, x_n) \to \min(\max) \qquad (9.3)$$

$$\left\{ h_i(x_1, x_2, \ldots, x_n) = b_i,\ i = \overline{1, m} \right\} \qquad (9.4)$$

Математикалық анализ курсында (9.3)-(9.4) есебін класикалық оңтайландыру есебі деп атайды.

Осы есептің шешімін табу үшін $\lambda_1,\lambda_2,\ldots,\lambda_m$ айнымалылар жиынын енгіземіз, олар Лагранж көбейткіштері деп аталады, және Лагранж функциясын құрайды:

$$L(x,\lambda) = f(x) + \sum_{i=1}^{m} \lambda_i [bi - h_i(x)] \quad . \qquad (9.5)$$

Дербес туындыларды табамыз

$$\partial L(x,\lambda)/\partial x_j \ (j = \overline{1,n}) \quad \text{және} \quad \partial L(x,\lambda)/\partial \lambda_i \ (i = \overline{1,m})$$

және m+n теңдеулер жүйесін қарастырамыз

$$\partial L(x,\lambda)/\partial x_j = \partial f(x)/\partial x_j - \sum_{i=1}^{m} \lambda_i \partial hi/\partial x_j = 0 (j = \overline{1,n}) \qquad (9.6)$$

$$\partial L(x,\lambda)/\partial \lambda_i = b_i - h_i(x) = 0 (i = \overline{1,m})$$

олардың m+n белгісіздік айнымалылары бар: $x_1, x_2, \ldots, x_n, \lambda_1, \lambda_2, \ldots, \lambda_n$.

(9.6) теңдеулер жүйесінің шешімі берілген функцияның барлық экстремум мәндерге ие бола алатын барлық нүктелерін береді. (9.3)-(9.4) есебінің экстремум нүктелерін анықтайтын әдіс Лагранждың көбейткіштер әдісі деп аталады.

Лагранждың көбейткіштер әдісінің алгоритмі:

1^0. Лагранж функциясын құру.

2^0. Лагранж функциясының x_j және λ_i айнымалылары бойынша дербес туындыларын тауып, оларды нөлге теңестіру.

3^0. (9.6) теңдеулер жүйесін шешіп, есептің мақсатты функциясының экстремумы болатындай нүктелерді табу.

4^0. Экстремумға күдікті нүктелердің арасынан экстремум алынатын нүктелерді тауып, осы нүктелердегі (9.3) функциясының мәнін есептеу.

19-мысал. Өнімді өндіру жоспары бойынша кәсіпорынға 180 бұйым дайындау керек. Бұл бұйымдарды екі технологиялық тәсілмен дайындауға болады. x_1 бұйымдарын І-ші тәсілмен өндіру кезіндегі шығын $4x_1+x_1^2$ ақша бірлігіне тең. Ал x_2 бұйымдарын ІІ-ші тәсілмен дайындау кезіндегі шығын $8x_2+x_2^2$ ақша бірлігіне тең. Өнімді өндіруге кететін жалпы шығындар минимал болатындай етіп, әрбір тәсілмен қанша бұйым дайындау керектігін анықтау.

Шешуі:

Есептің математикалық моделі

$f=4x_1+x_1^2+8x_2+x_2^2\rightarrow\min$,

$x_1+x_2=180$,

$x_1\geq 0$, $x_2\geq 0$.

Айнымалылардың теріс еместік талабын ескерместен, мақсатты функцияның минималды мәнін табайық. Ол үшін Лагранж функциясын құрамыз:

$$L(x,\lambda)=4x_1+x_1^{\,2}+8x_2+x_2^{\,2}+\lambda(180-x_1-x_2)$$

ғғғғғоларды нөлге теңестіреміз:

$4+2x_1-\lambda=0$,

$8+2x_2-\lambda=0$,

$180-x_1-x_2=0$.

Алғашқы екі теңдеудің λ-ларын оң жаққа өткізіп, олардың сол жақтарын теңестіре отырып, екі теңдеуден құралған жүйені аламыз:

$x_1-x_2=2$,

$180-x_1-x_2=0$.

Алатынымыз: $x_1^*=91$, $x_2^*=89$.

Бұл нүкте экстремумға күдікті нүкте болып табылады. Екінші дербес туындыларды қолдана отырып, мынаны аламыз: f функциясы осы нүктелерде шартты экстремумға ие.

9.5. Шартсыз оңтайландырудың бейсызықтық бағдарламалау есептері

Бұл параграфта туындыларды қолданбастан бір және көп өлшемді есептерді шешу әдістері қарастырылады.

9.5.1. Алтын қиық әдісі

Барлық бір өлшемді іздеу әдістері іздеу интервалын белгілі пропорциядағы екі кесіндіге бөлуге негізделеді. Алтын қиық әдісінде осы мақсатта Фибоначчи алтын сандары қолданылады:

$$\frac{1}{F_1}=\frac{F_1}{F_2},\quad F_1=0{,}62,\quad F_2=0{,}38\,.$$

Алтын қиық әдісінің алгоритмі келесідей:

1^0. f(x) мақсатты функциясы үшін шешімді іздеудің бастапқы интервалы таңдалады (анықталмағандық бөлігі) $a^{(0)} \le x \le b^{(0)}$, сондай-ақ шешімнің мүмкін дәлдігі $\Delta^{(0)}$ таңдалады.

2^0. $\Delta^{(k)}=b^{(k-1)} - a^{(k-1)}$. Егер $\Delta^{(k)} < \Delta^{(0)}$ болса, онда оңтайлы шешім табылды: $x^*= a^{(k-1)} + \Delta^{(k)}/2$, кері жағдайда 3^0-ші қадамға көшеміз.

3^0. $y_1^{(k)}= a^{(k-1)} + \Delta^{(k)}F_2$, $y_2^{(k)} = a^{(k-1)} + \Delta^{(k)}F_1$.

4^0. Егер $f(y_1^{(k)}) \le f(y_2^{(k)})$ болса, онда 5^0 қадамға көшеміз, кері жағдайда 6^0 қадамға көшеміз.

5^0. Кесіндінің оң жақ ұшын солға жылжытамыз $a^{(k)} = a^{(k-1)}$, $b^{(k)} = y_2^{(k)}$, 2^0 қадамға көшеміз.

6^0. Кесіндінің сол жақ ұшын оңға жылжытамыз $a^{(k)} = y_1^{(k)}$, $b^{(k)} = b^{(k-1)}$, 2^0 қадамға көшеміз.

20-мысал.

$f(x)=(x-3)^2+1 \rightarrow \min$, $1 \le x \le 4$, $\Delta=0{,}5$.

Жоғарыда көрсетілген әдіс алгоритмін қолданып, нәтижелерін келесі кестеге енгізейік.

110-кесте

x_1	x_2	Δ	y_1	y_2	$f(y_1)$	$f(y_2)$
1	4	3	2,14	2,86	1,7396	1,0966
2,14	4	1,86	2,85	3,28	1,0235	1,0802
2,14	3,28	1,14	2,57	2,85	1,1811	1,0328
2,57	3,28	0,71	3,84	3,01	1,0244	1,0002
3,84	3,28	0,44	3,04	3,12	1,0001	1,0135

Оңтайлы шешім: $x^*=3$, $f(x)=1$.

9.5.2. Циклдық әрбір координата бойынша құлдилау әдісі

Циклдық координаталық құлдилау әдісі туындыларды қолданбайтын көп өлшемді шартсыз оңтайландыру әдістеріне жатады. Көп өлшемділік дегеніміз әдістің бір айнымалыдан артық функцияның минимумын іздеу үшін қолданылуын білдіреді. Бұл әдіс минимумға қарай қозғалу бағытын анықтау үшін туындыларды қолданбайды, бірақ қозғалыс қадамын

анықтау үшін дербес туындыларды қолданады. Берілген әдіс релаксациялық әдіс болып табылады, яғни барлық айнымалылармен бірдей жүрмей, әрбір айнымалымен кезек-кезек жүреді, ол кезде басқаларының мәндері тұрақты болып табылады.

Циклдық координаталық құлдилау әдісінің алгоритмі келесідей:

1^0. n айнымалысы бар $f(x_1, x_2, ..., x_n)$ мақсатты функциясы үшін бастапқы вектор беріледі: $x^{(0)} = \begin{pmatrix} x_1^{(0)} \\ x_2^{(0)} \\ \vdots \\ x_n^{(0)} \end{pmatrix}$.

2^0. Бірлік вектор енгізіледі: $e_1 = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$, оның элементтерінің саны n-ге тең.

3^0. $f(x_1^{(0)} + \lambda_1 e_1,\ x_2^{(0)},\ ...,\ x_n^{(0)}) \to \min\limits_{\lambda_1}$ есебі шығарылады, ағымдық нүкте ретінде $x^{(1)} = \begin{pmatrix} x_1^{(1)} \\ x_2^{(0)} \\ \vdots \\ x_n^{(0)} \end{pmatrix}$ алынады.

4^0. Бірлік вектор енгізіледі: $e_2 = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix}$, оның элементтерінің саны n-ге тең.

5^0. $f(x_1^{(1)}, x_2^{(0)} + \lambda_2 e_2, ..., x_n^{(0)}) \to \min_{\lambda_2}$ есебі шығарылады, ағымдық нүкте ретінде $x^{(2)} = \begin{pmatrix} x_1^{(1)} \\ x_2^{(1)} \\ \vdots \\ x_n^{(0)} \end{pmatrix}$ алынады.

6^0. Қалған барлық айнымалылар бойынша кезекпен жүріп өтіледі, n-ші айнымалымен аяқтаймыз.

7^0. Барлық λ_i нөлге тең болғанша процес басынан бастап қайталанады; осы шарт орындалғанда, ағымдық X векторы оңтайлы болып табылады.

21-мысал.

$f(x) = x_1^{\,2} + 25x_2^{\,2} \to \min \quad x^0 = (2,2).$

Есептегі айнымалының екі координатасы болады. Бағыттың бірлік векторын бірінші координата бойынша енгіземіз. Вектордың өлшемі 2x1:

$1) e_1 = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$

$f(x_1^{(0)} - \lambda_0 e_1,\ x_2^{(0)}) \to \min_{\lambda_0}$

$(x_1^{(0)} - \lambda_0)^2 + 25x_2^{(0)}) \to \min_{\lambda_0}$

$-2(x_1^{(0)} - \lambda_0) = 0$

Берілген бағыт бойынша қозғалыс қадамын анықтаймыз:

$\lambda_0 = x_1^{(0)} = 2$

$x_1^{(1)} = x_1^{(0)} - \lambda_0 = 2 - 2 = 0$

Жаңа нүктенің координаталары:

$x^{(1)} = (0;2)$

$f(x) = 100$

Екінші координата бойынша бірлік векторды енгіземіз:

$2) e_2 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$

$(x_1^{(1)})^2 + 25(x_2^{(0)} - \lambda_1)^2 \to \min\limits_{\lambda_1}$

$-50(x_2^{(0)} - \lambda_1) = 0$

Берілген бағыт бойынша қозғалыс қадамы:

$\lambda_1 = x_2^{(0)} = 2$

$x_2^{(1)} = x_2^{(0)} - \lambda_1 = 2 - 2 = 0$

Жаңа нүктенің координаталары:

$x^{(2)} = (0;0)$

$f(x) = 0$

Оңтайлы шешім:

$x^* = (0;0)$

$f^*(x^*) = 0$

9.6. Туындыларды қолданатын көп өлшемді шартсыз оңтайландыру есептері

9.6.1. Барынша тез құлдилау әдісі

Есептің математикалық моделі берілген: $f(x) \to \min; X^{(0)}$.

Барынша тез құлдилау әдісі көп өлшемді оңтайландыру есептерін шешу үшін қолданылады. Алдыңғы әдістегідей, шешімді іздеуді бастау үшін қандай да бір бастапқы нүктені анықтау керек. Бірақ берілген әдістің айырмашылығы – ол қозғалыс бағытын анықтау үшін туындыларды қолданады. Берілген әдісті қолдану кезінде барлық координаталар бойынша бір уақытта жылжиды. Әдістің негізгі идеясы – функцияның барынша тез кему бағыты оның градиентіне қарама-қарсы бағытқа сәйкес келеді. Сондықтан функцияның градиентін тапқан соң бағыт ретінде оның модуліне тең, бірақ таңбасы қарама-қарсы мәнді таңдауға болады.

Барынша тез құлдилау әдісінің алгоритмі келесідей:

1^0. n айнымалысы бар f(x) функциясы үшін $x^{(0)}$ бастапқы нүктесі таңдалады, итерациялар санауышы k = 0.

2^0. Функцияның градиенті анықталады:

$$\nabla f(x^{(k)}) = \begin{pmatrix} \dfrac{\partial f(x)}{\partial x_1^{(k)}} \\ \vdots \\ \dfrac{\partial f(x)}{\partial x_n^{(k)}} \end{pmatrix}.$$

3^0. Эвклид нормасы есептеледі:

$$\left\|\nabla f(x^{(k)})\right\| = \sqrt{\left(\frac{\partial f(x)}{\partial x_1^{(k)}}\right)^2 + \ldots + \left(\frac{\partial f(x)}{\partial x_n^{(k)}}\right)^2}.$$

4^0. Тексеру: эвклид нормасы нөлге тең бе? Егер жауап «иә» болса, онда ағымдық нүкте оңтайлы болып табылады да, есептеулер тоқтатылады. Кері жағдайда 5^0 қадамға көшеміз.

5^0. Қозғалыс бағыты анықталады: $s^{(k)} = \begin{pmatrix} -\dfrac{\partial f(x)/\partial x_1^{(k)}}{\left\|\nabla f(x^{(k)})\right\|} \\ \vdots \\ -\dfrac{\partial f(x)/\partial x_n^{(k)}}{\left\|\nabla f(x^{(k)})\right\|} \end{pmatrix}$.

6^0. Табылған бағыт бойынша қадам шамасы анықталады: $f(x_1^{(k)} + \lambda_k s_1^{(k)}, ..., x_n^{(k)} + \lambda_k s_n^{(k)}) \to \min\limits_{\lambda_k}$.

7^0. Жаңа нүктенің координаталары анықталады: $x_j^{(k+1)} = x_j^{(k)} + \lambda_k s_j^{(k)}$, j = 1, …, n, k = k + 1. 2^0 қадамға көшеміз.

Барынша тез құлдилау әдісінде әдетте екі жағдай қарастырылады:

1) $\lambda^{(k)} - fix$; 2) $\lambda^{(k)} - \text{var}$.

22-мысал.

$f(x) = x_1^{\,2} + 25x_2^{\,2} \to \min, x^{(0)} = (2;2).$

1) λ қозғалыс қадамы тұрақты әрі 1-ге тең болатын жағдайды қарастырайық. Есепті барынша тез құлдилау әдісімен шығару нәтижелерін 111-кестеге енгізейік:

111-кесте

№ итер	x_1	x_2	df/dx_1	df/dx_2	$\|\nabla f\|$	s_1	s_2
0	2	2	4	100	100	-0,04	-1
1	1,96	1	3,92	50	50	-0,078	-1
2	1,88	0	3,76	0	3,76	-1	0
3	0,88	0	1,76	0	1,76	-1	0
4	-0,12	0	-0,24	0	0,24	1	0
5	0,88	0	1,76	0	1,76	-1	0

Оңтайлы шешім $X^{*\sim}(0.38; 0)$, $f^*(X^*)\tilde{}0{,}144$.

2) λ айнымалы шама болып табылатын жағдайды қарастырайық. Есепті барынша тез құлдилау әдісімен шығару нәтижелерін 112-кестеге енгізейік. 112-кестенің алдыңғы кестеден айырмашылығы – λ бағанының болуы. λ қозғалыс қадамын табуға арналған формуланы анықтау керек:

$$\lambda = -(s_1x_1+25s_2x_2)/((s_1)^2+25(s_2)^2)$$

112-кесте

№ итер	x_1	x_2	df/dx_1	df/dx_2	$\|\nabla f\|$	s_1	s_2	λ
0	2	2	4	100	100	-0,04	-1	-2
1	1,97	-0,003	3,84	-0,15	3,85	-0,99	-0,039	0,13
2	0,07	0,07	0,14	3,5	3,5	-0,04	-1	0,07
3	0,13	0	0,26	0	0,26	-1	0	0,13
4	0	0	0	0	0			

Оңтайлы шешім $X^*=(0; 0)$, $f^*(X^*)=0$.

9.6.2. Түйіндес градиент әдісі

Түйіндес градиент әдісі барынша тез құлдилау әдісіне қатты ұқсайды. Оның ерекшелігі – ол оңтайлы нүктеге қарай қозғалу бағыты ретінде градиентке қарама-қарсы бағытты емес, түйіндес бағыттарды қолданады, олар функцияның алдыңғы және ағымдық нүктелеріндегі градиенттерінің қатынасы негізінде есептеледі. Квадраттық бағдарламалау есептеріне қолданылатын мұндай амал n тәуелсіз айнымалысы бар функцияның оңтайлы шешімін n қадам ішінде табуға кепілдік береді.

Әдіс алгоритмі келесідей:

1^0. n айнымалысы бар f(x) функциясы үшін $x^{(0)}$ бастапқы нүктесі таңдалады, $\varepsilon > 0$ шешім дәлдігі, $k = 0$ итерациялар санауышы беріледі.

2^0. Функция градиенті анықталады $\nabla f(x^{(k)}) = \begin{pmatrix} \frac{\partial f(x)}{\partial x_1^{(k)}} \\ \vdots \\ \frac{\partial f(x)}{\partial x_n^{(k)}} \end{pmatrix}$.

3^0. Евклид нормасы анықталады $\left\|\nabla f(x^{(k)})\right\| = \sqrt{\left(\frac{\partial f(x)}{\partial x_1^{(k)}}\right)^2 + \ldots + \left(\frac{\partial f(x)}{\partial x_n^{(k)}}\right)^2}$, егер ол ε мәнінен аспайтын болса, демек ағымдық нүкте оңтайлы болып табылады да, есептеулер тоқтатылады. Кері жағдайда k = k + 1, 4^0 қадамға көшу.

4^0. Қозғалыс бағыты анықталады: егер k = 0 болса, онда $s^{(k)} = \begin{pmatrix} -\frac{\partial f(x)/\partial x_1^{(k)}}{\left\|\nabla f(x^{(k)})\right\|} \\ \vdots \\ -\frac{\partial f(x)/\partial x_n^{(k)}}{\left\|\nabla f(x^{(k)})\right\|} \end{pmatrix}$, кері жағдайда

$$s^{(k)} = -\nabla f(x^{(k)}) + s^{(k-1)} \frac{\nabla^{\mathrm{T}} f(x^{(k)}) \nabla f(x^{(k)})}{\nabla^{\mathrm{T}} f(x^{(k-1)}) \nabla f(x^{(k-1)})}.$$

5^0. Табылған бағыт бойынша қадам шамасы анықталады $f(x_1^{(k)} + \lambda_k s_1^{(k)}, ..., x_n^{(k)} + \lambda_k s_n^{(k)}) \to \min\limits_{\lambda_k}$.

6^0. Жаңа нүктенің координаталары анықталады $x_j^{(k+1)} = x_j^{(k)} + \lambda_k s_j^{(k)}$, j = 1, …, n, k = k + 1. 2^0 қадамға көшу.

9.6.3 Ньютон-Рабсон әдісі (екінші туындылар әдісі)

Бұл әдіс шектеулері жоқ квадраттық бағдарламалау есептерін шешу үшін қолданылады. Оңтайлы нүктені іздеу үшін бұл әдіс мақсатты функцияның екінші туындыларын қолданады.

Оңтайлы шешімді табу үшін әдістің есептеу процедураларының тек бір итерациясы ғана қажет.

Әдістің негізгі идеясы мынада: қандай да бір x^k нүктесінің төңірегіндегі екінші туындылары анықталған кез келген f(x) функциясын Тейлор қатарына жіктеуге болады:

$$f(x^{(k+1)}) = f(x^{(k)}) + \nabla^{\text{ò}} f(x^{(k)})(x^{(k+1)} - x^{(k)}) + \frac{1}{2} H(x^{(k)})(x^{(k+1)} - x^{(k)})^2,$$

мұндағы

$$H(x^{(k)}) = \begin{pmatrix} \dfrac{\partial^2 f(x^{(k)})}{\partial x_1^2} & \dfrac{\partial^2 f(x^{(k)})}{\partial x_1 \partial x_2} \\ \dfrac{\partial^2 f(x^{(k)})}{\partial x_1 \partial x_2} & \dfrac{\partial^2 f(x^{(k)})}{\partial x_2^2} \end{pmatrix}$$ – Гессе (гессиан) матрицасы, ол f(x) функциясының x_k нүктесіндегі екінші дербес туындыларынан тұрады.

Тейлор қатарына жіктелген функцияның минимумын табу үшін оның $(x^{(k+1)} - x^{(k)})$ бойынша туындысын табамыз, сондай-ақ $x^{(k+1)}$-ті өрнектейміз, нәтижесінде мынаны аламыз:

$$x^{(k+1)} = x^{(k)} - H^{-1}(x^{(k)}) \nabla f(x^{(k)}).$$

Осылайша, біз бастапқы нүктеден бірден оңтайлы нүктеге апаратын формуланы аламыз.

Ньютон-Рабсон әдісінің алгоритмі:

1^0. n айнымалысы бар f(x) функциясы үшін $x^{(0)}$ бастапқы нүктесі таңдалады.

2^0. Функция градиенті анықталады $\nabla f(x^{(k)})$.

3^0. Гессе матрицасы анықталады $H(x^{(k)})$.

4^0. Кері матрица анықталады $H^{-1}(x^{(k)})$.

5^0. Оңтайлы нүкте анықталады $x^{(k+1)} = x^{(k)} - H^{-1}(x^{(k)}) \nabla f(x^{(k)})$.

23-мысал

Келесі түрдегі бейсызықтық функция берілген:

$$9x_1^2 + 16x_2^2 - 90x_1 - 128x_2 \to \min$$

бастапқы нүктесі $x^{(0)} = (0;0)$. Оңтайлы нүктені анықтау керек.

Шешуі:

$$f(x^{(0)}) = 0$$

Градиенттерді анықтаймыз:

$$\frac{\partial f}{\partial x_1} = 18x_1 - 90\Big|_{x_1 = 0} = -90$$

$$\frac{\partial f}{\partial x_2} = 32x_2 - 128\Big|_{x_2 = 0} = -128$$

Демек,

$$\nabla f(\vec{x}^{(0)}) = \begin{vmatrix} -90 \\ -128 \end{vmatrix}$$

Гессе матрицасын табамыз:

$$H(x) = \begin{vmatrix} 18 & 0 \\ 0 & 32 \end{vmatrix}$$

Кері матрицаны анықтаймыз:

$$H^{-1}(x) = \frac{1}{576}\begin{vmatrix} 32 & 0 \\ 0 & 18 \end{vmatrix} = \begin{vmatrix} \frac{1}{18} & 0 \\ 0 & \frac{1}{32} \end{vmatrix}.$$

Оңтайлы нүкте:

$$x^{(1)} = \begin{vmatrix} 0 \\ 0 \end{vmatrix} - \begin{vmatrix} \frac{1}{8} & 0 \\ 0 & \frac{1}{32} \end{vmatrix} \begin{vmatrix} -90 \\ -128 \end{vmatrix} = \begin{vmatrix} 0 \\ 0 \end{vmatrix} - \begin{vmatrix} -5 \\ -4 \end{vmatrix} = \begin{vmatrix} 5 \\ 4 \end{vmatrix}$$

Жауабы: $x^* = x^{(1)} = (5;4)$

$$f(x^{(0)}) = 225 + 256 - 450 - 512 = -481.$$

9.7. Шартты оңтайландырудың көп өлшемді бейсызықтық есептері

9.7.1. Франк-Вульф әдісі

Франк-Вульф әдісі шектеулері бар квадраттық бағдарламалау есептерін шешу үшін қолданылады.

Есептің математикалық моделі берілген:

$$f(x) = \sum_{j=1}^{n} c_j x_j + \sum_{j=1}^{n}\sum_{i=1}^{n} d_{ij} x_i x_j \to \max \qquad (9.7)$$

$$\sum_{j=1}^{n} a_{ij} x_j \ge b_i, i = \overline{1,m}, x_j \ge 0, j = \overline{1,n} \qquad (9.8)$$

Франк-Вульф алгоритмінің алгоритмі:

1^0. Симплекс әдісінің көмегімен $x^{(0)}$ бастапқы нүктесі анықталады, итерациялар санауышы k=0.

2^0. Қозғалыс бағыты анықталады $s^{(k)} = x^{(*)} - x^{(k)}$, мұндағы $x^{(*)}$ – бастапқы еепте берілген шектеулері бар және төмендегідей мақсатты функциясы бар сызықтық бағдарламалау есебінің шешімі:

$$\sum_{j=1}^{n} \frac{\partial f}{\partial x_j}(x_j - x_j^{(k)}) \to \max .$$

Есеп симплекс әдісімен шығарылады.

3^0. Қозғалыс қадамын анықтау

$$f(x_1^{(k)} + \lambda^* s_1^{(k)}, ..., x_n^{(k)} + \lambda^* s_n^{(k)}) \to \max_{\lambda^*},$$

$$\lambda^{(k)} = \min\{1; \lambda^*\}.$$

4^0. Жаңа нүктенің координаталары анықталады $x_j^{(k+1)} = x_j^{(k)} + \lambda^{(k)} s_j^{(k)}$, j = 1, …, n.

5^0. Шартты тексеру: егер $f(x^{(k+1)}) > f(x^{(k)})$ болса, онда k = k + 1 және 2^0 қадамға өту, кері жағдайда $x^{(k)}$ – оңтайлы нүкте.

24-мысал.

$f(x)=2x_1-3x_2-x_1^2\rightarrow max$

$x_1+2x_2\leq 16$

$5x_1+2x_2\leq 40$

$x_1\geq 0,\ x_2\geq 0$

Шешуі:

$df/dx=2-3-2x_1\rightarrow max$

$df/dx=-1-2x_1\rightarrow max$

113-кесте

	$-x_1$	$-x_2$	1
y_1	1	2	16
y_2	5	2	40
Z	2	0	-1

$x^{(0)}=(0; 0)$

$s^{(k)}=x^{(*)}-x^{(0)}$

$$\sum_{j=1}^{2} df/dx(x_j - x_j^{(0)} \rightarrow \max$$

$df/dx_1=2-2x_1|_{x1=0}=2$

$df/dx_2=-3|_{x2=0}=-3$

$2(x_1-0)-3(x_2-0)\rightarrow max$

$2x_1-3x_2\rightarrow max$

114-кесте

	$-x_1$	$-x_2$	1
y_1	1	2	16
y_2	5	2	40
Z	-2	3	0

115-кесте

	$-y_2$	$-x_2$	1
y_1	-1/5	8/5	8
x_1	1/5	2/5	8
Z	2/5	19/5	16

$x^{(*)}=(8; 0)$

$s_1^{(0)}=8,\ s_2^{(0)}=0$

$f(x)=2(x_1^{(0)}+\lambda^* s_1^{(0)})-3(x_2^{(0)}+\lambda^* s_2^{(0)})-(x_1^{(0)}+\lambda^* s_1^{(0)})^2\to\max$
$f(x)=16\lambda^*-64(\lambda^*)^2\to\max$
$16-128\lambda^*=0$
$\lambda^*=16/128=1/8 \quad \lambda^0=\min[1;1/8]=1,8.$

$x^{(1)}=(1,0)$
$f(x^{(0)})=0$
$s^{(1)}=x^*-x^{(1)}$
$f(x^{(1)})=1.$

$df/dx_1=2-2x_1|_{x1=1}=0$
$df/dx_2=-3$
$z=-3(x_2-0)\to\max$
$x^*=(0,0) \quad s^{(1)}=(-1,0).$

$f(x)=2(x_1^{(1)}+\lambda^* s_1^{(1)})-3(x_2^{(1)}+\lambda^* s_2^{(1)})-(x_1^{(1)}+\lambda^* s_1^{(1)})^2\to\max$
$2(1-\lambda^*)-3(0+0)-(1-\lambda^*)^2\to\max$
$-2+2(1-\lambda^*)=0$
$-2+2-2\lambda^*=0$
$\lambda^*=0,\ \lambda^{(1)}=1.$

Есептің шешуін кестеге енгізейік:

116-кесте

№	x_1	x_2	f(x)	df/dx_1	df/dx_2	s_1	s_2	λ^*	$\lambda_к$
0	0	0	0	2	-3	8	0	1/8	1/8
1	1	0	1	0	-3	-1	0	0	0
2	1	0	1						

$x^{(2)}=(1;0)$ – оңтайлы нүкте.

9.7.2. Зойтендейктің мүмкін бағыттар әдісі

Берілген әдіс жалпы түрдегі дөңес бағдарламалау есептеріне қолданылады:

$$z = f(x) \to \min$$

$$\varphi_i(x) \le 0, \quad i = \overline{1,m}.$$

f(x) және $\varphi_i(x)$ функцияларын $x^{(k)}$ нүктесінің маңайында Тейлор қатарына жіктейміз:

$$f(x) = f\left(x^{(k)}\right) + \nabla^{\mathrm{T}} f\left(x^{(k)}\right)\left(x - x^{(k)}\right) \to \min, \qquad (9.9)$$

$$\varphi_i(x) = \varphi_i\left(x^{(k)}\right) + \nabla^{\mathrm{T}} \varphi_i\left(x^{(k)}\right)\left(x - x^{(k)}\right) \le 0, \quad i = \overline{1,m}. \qquad (9.10)$$

$x^{(k)}$ нүктесінен $x^{(k+1)}$ нүктесіне ауысу $x^{(k+1)} = x^{(k)} + \lambda^{(k)} s^{(k)}$ қатынасымен беріледі. (9.9)-(9.10) есепке x векторының орнына оның ағымдық $x^{(k+1)}$ мәнін қойып, келесі есепті аламыз:

$$f(x) = f\left(x^{(k)}\right) + \lambda^{(k)} \nabla^{\mathrm{T}} f\left(x^{(k)}\right) s^{(k)} \to \min, \qquad (9.11)$$

$$\varphi_i(x) = \varphi_i\left(x^{(k)}\right) + \lambda^{(k)} \nabla^{\mathrm{T}} \varphi_i\left(x^{(k)}\right) s^{(k)} \le 0, \quad i = \overline{1,m}. \qquad (9.12)$$

$f\left(x^{(k)}\right)$ және $\varphi_i\left(x^{(k)}\right)$ тұрақты болғандықтан, $x^{(k)}$ нүктесінен $x^{(k+1)}$ нүктесіне ауысудың қажетті және жеткілікті шарттары келесі түрде болады:

$$\lambda^{(k)} \nabla^{\mathrm{T}} f\left(x^{(k)}\right) s^{(k)} < 0, \qquad (9.13)$$

$$\lambda^{(k)} \nabla^{\mathrm{T}} \varphi_i\left(x^{(k)}\right) s^{(k)} \le 0, \quad i = \overline{1,m}. \qquad (9.14)$$

(9.13)-(9.14) теңсіздіктерін қанағаттандыратын кез келген $s^{(k)}$ векторы мүмкін бағыттағы вектор болып табылады.

Зойтендейктің мүмкін бағыттар әдісінің алгоритмі:

1^0. $x^{(0)}$ бастапқы шешімін анықтау. Егер есеп сызықтық бағдарламалау есебі болса, симплекс әдісімен, егер бейсызықтық болса, сәйкестенбеу әдісімен анықтаймыз.

2^0. Берілген $x^{(k)}$ бойынша (9.13)-(9.14) есептерін шешу:

$$\lambda^{(k)} \nabla^{\mathrm{T}} f\left(x^{(k)}\right) s^{(k)} \to \min, \qquad (9.15)$$

$$\lambda^{(k)} \nabla^{\mathrm{T}} \varphi_i\left(x^{(k)}\right) s^{(k)} \le 0, \quad i = \overline{1,m}. \qquad (9.16)$$

және $s^{(k)}$ табу.

3^0. Шартты тексеру: егер $\lambda^{(k)}\nabla^{\mathrm{T}} f\left(x^{(k)}\right)s^{(k)} < 0$ болса, онда 4^0 қадамға көшеміз, кері жағдайда оңтайлы шешім табылды: $x^* = x^{(k)}$.

4^0. Алдымен $\lambda^* = \max\left\{\lambda \middle| \varphi_i\left(x^{(k)} + \lambda^* s^{(k)}\right) \le 0,\ \ i = \overline{1,m}\right\}$ есебін шешу, содан кейін $f\left(x^{(k)} + \lambda^* s^{(k)}\right) \to \min\limits_{0 \le \lambda^{(k)} \le \lambda^*}$ есебін шешу арқылы $\lambda^{(k)}$ табу.

5^0. Жаңа нүктенің координаталарын анықтау $x^{(k+1)} = x^{(k)} + \lambda^{(k)} s^{(k)}$. 2^0 қадамға көшу.

Сұрақтар мен тапсырмалар

1. Бейсызықтық бағдарламалау есебінің математикалық қойылымын келтіру.
2. Бейсызықтық қос мағыналы есепті құрудың негізгі ережелерін келтіру.
3. Бейсызықтық бағдарламалаудың қос мағыналылық теоремасын тұжырымдау.
4. Бейсызықтық бағдарламалау әдістерінің жіктелуін келтіру.
5. Алтын қиық әдісі қандай есептерге қолданылады?
6. Барынша тез құлдилау әдісі қандай есептерге қолданылады?
7. Франк-Вульф әдісі қандай есептерге қолданылады?
8. Зойтендейктің мүмкін бағыттар әдісі қандай есептерге қолданылады?
9. Келесі есепті алтын қиық әдісімен шығару:
   $f(x) = x2 - 5x + 10 \to \min$, $1 \le x \le 7$.
10. Келесі есепті циклдық әрбір координата бойынша құлдилау әдісімен шығару:
    $z = 4x_1 + 8x_2 - 2x_1^{\ 2} - 2x_2^{\ 2}(\max),\ x^{(0)} = (5,10)$.
11. Келесі есепті барынша тез құлдилау әдісімен шығару:
    $z = (x_1 - 2)^2 + (x_2 - 1)^2, x^{(0)} = (4;5)$.
12. Келесі есепті Ньютон-Рабсон әдісімен шығару:
    $z = 20x_1 + 16x_2 - x_1^{\ 2} - x_2^{\ 2}(\max),\ x^{(0)} = (4,4)$
13. Келесі есепті Франк-Вульф әдісімен шығару:
    $f(x)=10-2x_1+x_2-x_1^{\ 2}-x_2^{\ 2}$ (max),

$$\left.\begin{array}{l} 4x_1+5x_2\le 80 \\ 2x_1+x_2\le 34 \\ x_1\ge 0,\ x_2\ge 0. \end{array}\right\},$$

14. Лагранждың көбейткіштер әдісін қолдана отырып, функцияның шартты экстремумдарын табу:

$$f(x)=3x_1^2+2x_1+2x_2^2+4x_1x_2\to \min,$$

$$\left.\begin{array}{l} x_1^2+2x_2^2=19 \\ x_1+2x_2x_3=11 \end{array}\right\}.$$

15. Келесі есепті барынша тез құлдилау әдісін қолданып шығару керек:

$$f(x)=6x_1+32x_2-x_1^2-4x_2^2\to \max$$

$$X^{(0)}=(0,0).$$

16. Келесі есепті барынша тез құлдилау әдісін қолданып шығару керек:

$$f(x)=x_1^2+x_2^2-6x_1-4x_2\to \min$$

$$X^{(0)}=(0,0).$$

17. Келесі есепті барынша тез құлдилау әдісін қолданып шығару керек:

$$f(x)=2x_1-x_1^2-x_2\to \min$$

$$X^{(0)}=(1,2).$$

18. Келесі есептерді Франка-Вулф әдісімен шығару керек:

$$f(x)=10+16x_2-x_1^2-x_2^2\to \max$$

$$x_1+2x_2\le 16$$

$$5x_1+2x_2\le 40$$

$$x_1\ge 0, x_2\ge 0$$

19. Келесі есептерді Франка-Вулф әдісімен шығару керек:

$$f(x)=x_1^2+x_2^2-6x_1+6x_2\to \max$$

$$x_1\le 6$$

$$x_2\le 2$$

$$x_1\ge 0$$

$$x_2\ge 0.$$

20. Келесі есептерді Франка-Вулф әдісімен шығару керек:

$$f(x) = x_1 + 2x_1^2 + 2x_2 + 4x_3 \to \min$$
$$x_1 + 2x_2 + 3x_3 \le 15$$
$$3x_1 + x_2 + x_3 \le 12$$
$$x_1 \ge 0, x_2 \ge 0, x_3 \ge 0$$
$$X^0 = (5, 4, 2)$$

21. Келесі есептерді Франка-Вулф әдісімен шығару керек:

$$f(x) = 10x_2 - 6x_1^2 - 19x_2^2 + 16x_1x_2 \to \max$$
$$3x_1 + x_2 \le 3$$
$$x_1 \ge 0,\ x_2 \ge 0.$$

22. Келесі есептерді Франка-Вулф әдісімен шығару керек:

$$f(x) = -x_1^2 - x_2^2 \to \max$$
$$x_1 + 0{,}5x_2 \ge 1$$
$$x_1 + 0{,}5x_2 \le 4$$
$$x_1 + x_2 \le 6$$
$$x_1 \ge 0,\ x_2 \ge 0.$$

23. Келесі есептерді Франка-Вулф әдісімен шығару керек:

$$f(x) = 4x_1 + 10x_2 - x_1^2 - x_2^2 \to \max$$
$$x_1 + x_2 \le 4$$
$$x_2 \le 2$$
$$x_1 \ge 0,\ x_2 \ge 0.$$

10. ОЙЫНДАР ЖӘНЕ СТАТИСТИКАЛЫҚ ШЕШІМДЕР ТЕОРИЯСЫ

10.1. Негізгі ұғымдар мен анықтамалар

Ойындар теориясы – бұл математикалық модельдер теориясы, ондағы қатысушылардың мүдделері әр түрлі болады, және де олар өз мақсаттарына түрлі жолдармен жетеді. Қатысушылардың қарама-қарсы мүдделерінің қақтығысы жанжалды жағдайдың туындауына алып келеді.

Ойындар теориясы жанжалды жағдайда шешім қабылдаудың математикалық теориясы болып табылады.

Жанжалды жағдайдың жеңілдетілген формалданған моделі ойын деп аталады, ал жанжалдасушы тараптар ойыншылар деп аталады.

Ойындар теориясындағы жүріс деп ойын ережелерінде алдын ала ескерілген әрекеттердің бірін таңдап, оны жүзеге асыруды айтамыз. Жүрістер жеке және кездейсоқ болады.

Жеке жүріс деп ойыншының әрекеттердің бірін саналы түрде таңдап, оны жүзеге асыруын айтады.

Кездейсоқ жүріс деп бірқатар мүмкіншіліктер ішінен ойыншының емес, қандай да бір кездейсоқ таңдау механизмінің (тиын лақтыру, карта үлестіру) таңдауын айтады.

Ойыншының стратегиясы деп ойын барысында қалыптасқан жағдайға байланысты ойыншының әрбір жеке жүрісі кезіндегі әрекеттер нұсқасын анықтайтын ережелер жиынтығын айтады.

Стратегиялардың санына байланысты ойындар шекті және шексіз болып бөлінеді, ал ойыншылардың санына байланысты жұп ойындар және жиындалған ойындар болып бөлінеді.

Ойындар теориясының мақсаты әрбір ойыншы үшін оңтайлы стратегияны, яғни минимал ұтылыс немесе максимал орташа ұтыс қамтамасыз етілетіндей стратегияны анықтау болып табылады.

Ойынның ең қарапайым түрі – қосындысы нөлге тең екі адамның ойыны, олардың ұтыстарының қосындысы нөлге тең болады. А ойыншысы A_i, i =1, …, m, мүмкін стратегиялардың

арасынан біреуін таңдайды. В ойыншысы B_j, j=1,…,n, мүмкін стратегиялардың арасынан біреуін таңдайды. Ойыншылар әрбір стратегияны басқа ойыншының таңдауын білмей тұрып таңдайды.

Әрбір ойын үшін төлем матрицасын құруға болады:

B_j / A_i	B_1	B_2	…	B_n
A_1	a_{11}	a_{12}	…	a_{1n}
A_2	a_{21}	a_{22}	…	a_{2n}
…	…	…	…	…
A_m	a_{m1}	a_{m2}	…	a_{mn}

,

мұндағы a_{ij} – бірінші А ойыншысының ұтысы (В ойыншысының ұтылысы), егер А ойыншысы A_i стратегиясын таңдаған болса (егер В ойыншысы B_j стратегиясын таңдаған болса).

Кепілденген нәтиже принципі:

А ойыншысы A_i стратегиясын таңдады делік, онда ең нашар жағдайларда ол $\alpha_i = \min\limits_j a_{ij}$ ұтысын алады.

Осындай мүмкіндікті алдын-ала көре отырып, А ойыншысы өзінің минимал ұтысын максималдайтындай стратегияны таңдайды $\alpha = \max\limits_i \min\limits_j a_{ij}$.

α – А ойыншысының кепілденген ұтысы (ойынның ең төмен бағасы).

α жағдайын қамтамасыз ететін A_{i_0} стратегиясы максиминді стратегия деп аталады.

В ойыншысы стратегияны таңдаған кезде келесі принципті негізге алады. Ізделініп отырған B_j стратегиясын таңдау кезінде оның ұтылысы j-ші бағандағы $\beta_j = \max\limits_i a_{ij}$ максималды мәндерінен аспайды.

В ойыншысы өзінің ұтылысы минимал болатындай стратегияны таңдайды $\beta = \min\limits_j \max\limits_i a_{ij}$.

β – ойынның ең жоғары бағасы ($\alpha \le \nu \le \beta$).

β жағдайын қамтамасыз ететін B_{j_0} стратегиясы минимаксты стратегия деп аталады.

Егер $\alpha = \beta$ болса, онда шешім таза стратегияларда анықталады, ал қалған жағдайларда аралас стратегияларда анықталады.

Аралас стратегия – бұл ойыншының өз стратегияларын кездейсоқ таңдауы.

$S_A = (p_1, p_2, \ldots, p_m)$ – А ойыншысының аралас стратегиясы, мұндағы $A_1, A_2, \ldots, A_m$ стратегиялары $p_1, p_2, \ldots, p_m$ ықтималдықтарымен қолданылады, және $\sum_{i=1}^{m} p_i = 1$.

$S_B = (q_1, q_2, \ldots, q_n)$ – В ойыншысының аралас стратегиясы, мұндағы $B_1, B_2, \ldots, B_n$ стратегиялары $q_1, q_2, \ldots, q_n$ ықтималдықтарымен қолданылады, және $\sum_{j=1}^{n} q_j = 1$.

А және В ойыншыларының стратегиялары, $p_i > 0$ және $q_j > 0$ болған жағдайда, белсенді стратегиялар деп аталады.

Аралас стратегиялар кезіндегі А ойыншысының ұтысы ұтыстың математикалық күтімі ретінде анықталады:

$$\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij} p_i q_j = S_A S_B^T .$$

Ойындар теориясының негізгі теоремасы. Әрбір шекті ойынның кем дегенде бір шешімі болады, ол аралас стратегиялар облысында болуы мүмкін.

Ойыншылардың оңтайлы стратегиясы үшін мына қатынас орын алады:

$$\nu = \max_{S_A} \min_{S_B} S_A S_B^T A = \min_{S_B} \max_{S_A} S_A S_B^T A,$$

мұндағы ν – ойынның бағасы.

А ойыншысының S_A^* оңтайлы стратегиясын қолдануы нәтижесінде ол В ойыншысының кез келген әрекетінде ойын бағасынан төмен емес ұтысты алуы тиіс:

$$\sum_{i=1}^{m} a_{ij} p_i^* \geq \nu , \quad j = \overline{1,n} .$$

В ойыншысы үшін де солай: В ойыншысының S_B^* оңтайлы стратегиясын қолдануы нәтижесінде ол А ойыншысының кез келген әрекетінде ойын бағасынан жоғары емес ұтысты алуы тиіс:

$$\sum_{j=1}^{n} a_{ij} q_j^* \leq \nu , \; i = \overline{1,m} .$$

10.2. Екілік сипаттау әдісі (алгебралық әдіс)

Төлем матрицасы келесі түрде берілсін:

	B_1	B_2
A_1	a_{11}	a_{12}
A_2	a_{21}	a_{22}

А ойыншысы үшін стратегия $X = (x_1, x_2)$ векторымен, В ойыншысы үшін – $Y = (y_1, y_2)$ векторымен беріледі.

Сонда В ойыншысы таза B_1 немесе B_2 стратегиясын таңдаған кезде, А ойыншысы ойын бағасына тең орташа ұтысты алады:

$$A: \begin{cases} a_{11}x_1 + a_{21}x_2 = \nu, \\ a_{12}x_1 + a_{22}x_2 = \nu, \\ x_1 + x_2 = 1, \end{cases}$$

В ойыншысы үшін де осылай:

$$B: \begin{cases} a_{11}y_1 + a_{12}y_2 = \nu, \\ a_{21}y_1 + a_{22}y_2 = \nu, \\ y_1 + y_2 = 1. \end{cases}$$

Теңдеулер жүйесін шеше отырып, екі ойыншы үшін де оңтайлы аралас стратегияларды табамыз: $X^* = (x_1^*, x_2^*)$, $Y^* = (y_1^*, y_2^*)$.

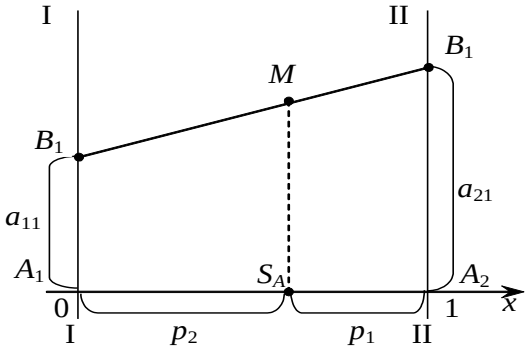
10.3. 2 × 2 ойынын шешудің графикалық әдісі

Төлем матрицасындағы сандар оң таңбалы болуы керек, немесе оларды сондай түрге келтіру керек, яғни бүкіл төлем матрицасын қандай да бір оң санды қосу керек.

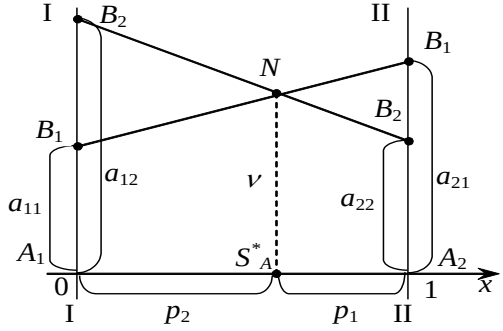
Абсциссалар осінің ұзындығы 1-ге тең кесіндісін аламыз (25-сурет). Кесіндінің сол жақ ұшы (абсциссасы x = 0 болатын нүкте) A_1 стратегиясына, оң жақ ұшы (x = 1) A_2 стратегиясына сәйкес келеді; кесіндінің барлық аралық нүктелері А ойыншысының аралас стратегияларына сәйкес келеді, және де A_1 стратегиясының p_1 ықтималдығы S_A нүктесінен бастап кесіндінің соңына дейінгі аралыққа тең болады, ал A_2 стратегиясының p_2 ықтималдығы кесіндінің сол жақ ұшына дейінгі ара қашықтыққа тең болады. A_1 және A_2 нүктелері арқылы абсциссалар осіне перпендикуляр жүргізейік: I – I осі және II – II осі. I – I осіне A_1 стратегиясында алынған ұтыстарды, ал II – II осіне A_2 стратегиясында алынған ұтыстарды қоямыз.

В ойыншысы B_1 стратегиясын қолдансын; ол I – I және II – II осьтерінде ординаталары сәйкесінше a_{11} және a_{21} болатын нүктелерді береді. Осы нүктелер арқылы B_1B_1 түзуін жүргізейік. Кез келген $S_A = (p_1, p_2)$ аралас стратегиясында біздің ұтыс B_1B_1 түзуінің бойындағы М нүктесімен өрнектелетіні анық, бұл нүкте кесіндіні $p_2 : p_1$ қатынасында бөлетін абсциссалар осіндегі нүктеге сәйкес келеді. B_1B_1 түзуін шартты түрде «B_1 стратегиясы» деп атайық.

Дәл осындай тәсілмен B_2 стратегиясын да құруға болатыны анық (26-сурет).



25-сурет

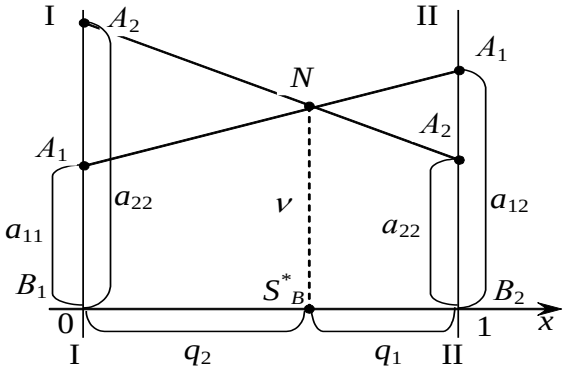


26-сурет

Бізге S^*_A оңтайлы стратегиясын табу керек, яғни біздің минимал ұтысымыз (біз үшін В-ның ең нашар әрекетінде) максимумға айналатындай стратегия. Ол үшін B_1, B_2 стратегияларындағы ұтыстың ең төменгі шекарасын, яғни B_1NB_2 сынық сызығын сызайық. Бұл шекарада А ойыншысының кез келген аралас стратегиясындағы минимал ұтысы жататын болады. Осы ұтыс максимал мәнге жететін N нүктесі ойынның шешімі мен бағасын анықтайды. Осыдан көрініп тұрғандай, N нүктесінің ординатасы ν ойын бағасы болып табылады, ал оның абсциссасы p_2-ге тең, ал кесіндінің оң жақ ұшына дейінгі ара қашықтық p_1-ге тең, яғни S^*_A нүктесінен бастап кесіндінің ұштарына дейінгі ара қашықтық А ойыншысының оңтайлы

аралас стратегиясында A_2 және A_1 стратегияларының p_2 және p_1 ықтималдықтарына тең.

S^*_B = (q_1, q_2) оңтайлы стратегиясы осындай тәсілмен табылады, ол үшін А және В ойыншыларының орнын ауыстыру керек, ал ұтыстың төменгі шекарасының максимумының орнына жоғарғы шекараның минимумын қарастыру керек (27-сурет).



27-сурет

10.4. Жиындалған ойындарға графикалық әдісті қолдану

2xn немесе mx2 ойынын шешу әдісінің алгоритмі:

1^0. Екінші (бірінші) ойыншының стратегиясына сәйкес келетін түзулер сызылады.

2^0. Ұтыстың төменгі (жоғарғы) шекарасы анықталады.

3^0. Екінші (бірінші) ойыншының екі стратегиясы табылады, оларға максималды (минималды) ординатасы бар нүктеде қиылысатын екі түзу сәйкес келеді.

4^0. Оңтайлы стратегиядағы ойын бағасы анықталады.

25-мысал. Төмендегідей төлем матрицасымен берілген ойынның шешімін табу:

$$A = \begin{pmatrix} 8 & 5 & 3 & 6 & 7 \\ 4 & 7 & 9 & 5 & 8 \end{pmatrix}.$$

Шешуі. Есепті шешу графигі 28-суретте келтірілген.

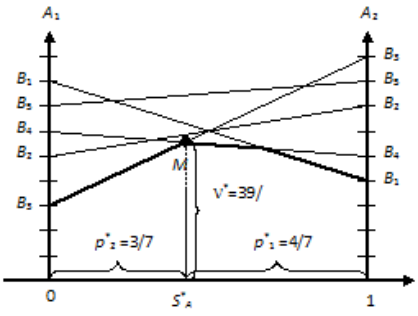
$$\nu^* = 39/7; \rho_1^* = 4/7; p_2^* = 3/7; p_1 + p_2 = 1.$$

Осылайша m × 2 матрицасын да шешуге болады. Тек бұл жағдайда ойынның жоғарғы шекарасы сызылады да, ондағы минималды нүкте анықталады (29-сурет).

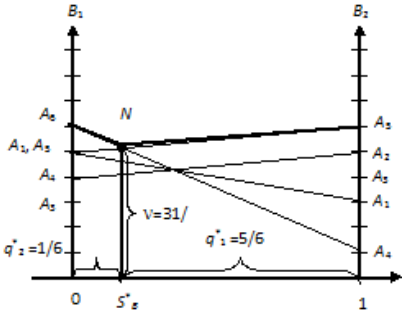
26-мысал. Төлем матрицасында теріс элементтер бар, сондықтан бүкіл матрицаға 4 санын қосамыз.

$$A = \begin{pmatrix} 1 & -1 \\ 0 & 1 \\ -1 & 0 \\ 2 & -3 \\ 1 & 2 \end{pmatrix} + 4 \Rightarrow \quad A = \begin{pmatrix} 5 & 3 \\ 4 & 5 \\ 3 & 4 \\ 6 & 1 \\ 5 & 6 \end{pmatrix}.$$

$$\nu^* = 5\frac{1}{6} - 4 = 1\frac{1}{6}; q_1^* = 5/6; q_2^* = 1/6; q_1^* + q_2^* = 1.$$



26-сурет



27-сурет

10.5. Ойындар теориясының есептерін сызықтық бағдарламалау есептеріне келтіру

А ойыншысының m стратегиясы бар болсын: A_1, A_2, …, A_m, ал В ойыншысының n стратегиясы бар болсын: B_1, B_2, …, B_n. Келесі түрдегі төлем матрицасы берілген

B_j / A_i	B_1	B_2	…	B_n
A_1	a_{11}	a_{12}	…	a_{1n}
A_2	a_{21}	a_{22}	…	a_{2n}
…	…	…	…	…
A_m	a_{m1}	a_{m2}	…	a_{mn}

,

Аралас стратегиялардағы ойынның шешімін табу керек:

$$S_A^* = \left(p_1^*,\ p_2^*,\ ...,\ p_m^*\right), \qquad \sum_i p_i = 1,$$

$$S_B^* = \left(q_1^*,\ q_2^*,\ ...,\ q_n^*\right), \qquad \sum_j q_j = 1.$$

Ойындар теориясының негізгі теоремасына сәйкес А ойыншысы үшін және В ойыншысы үшін қосы мағыналы есептер жұбын жазамыз:

A	B
$\nu \to \max$	$\nu \to \min$
$a_{11}p_1 + a_{21}p_2 + ... + a_{m1}p_m \geq \nu$	$a_{11}q_1 + a_{12}q_2 + ... + a_{n1}q_n \leq \nu$
.....	.....
$a_{1n}p_1 + a_{2n}p_2 + ... + a_{mn}p_m \geq \nu$	$a_{m1}q_1 + a_{m2}q_2 + ... + a_{mn}q_n \leq \nu$
$\sum_i p_i = 1,\ u_i = \frac{p_i}{\nu}$	$\sum_j q_j = 1,\ \xi_j = \frac{q_j}{\nu}$

Анықтылық үшін, ν>0 деп алайық. А матрицасының барлық элементтеріне тұрақты С санын қосқан жағдайда

оңтайлы стратегиялардың өзгермейтіндігінің, тек ойын бағасын С-ға арттыратынының арқасында осылай ала беруімізге болады.

Септі А ойыншысы үшін қарастырайық: теңсіздік-шектеулердің екі жағын да v-ға бөлейік, сонда мынаны аламыз

$$\sum_{i=1}^{m} a_{ij} \frac{u_i^*}{v} \geq 1, (j = \overline{1,n}).$$

$u_i^* = p_i^*/v$, деп алайық, сонда

$$\sum_{i=1}^{m} a_{ij}\, u_i^* \geq 1 \left(j = \overline{1,n}\right);\ u_i^* \geq 0 \left(i = \overline{1,m}\right).$$

Енгізілген белгілеуді қолдана отырып, шартты қайта жазайық

$\sum_{i=1}^{m} p_i^* = 1$ өрнегін мына түрде $\sum_{i=1}^{m} u_i^* = 1/v$.

Бірінші ойыншы максималды ұтысты алуға ұмтылғандықтан, ол 1/v шамасына минимумды қамтамасыз етуі тиіс. Осыны ескере отырып, бірінші ойыншының оңтайлы стратегиясының анықталуы төмендегі функцияның минимал мәнін табуы арқылы жүзеге асырылады:

$$\omega = \sum_{i=1}^{m} u_i \rightarrow \min$$

мынадай шарттар орындалуы тиіс

$$\sum_{i=1}^{m} a_{ij}\, u_i \geq 1 \left(j = \overline{1,n}\right);\ u_i \geq 0 \left(i = \overline{1,m}\right).$$

Осыған ұқсас пайымдаулар бойынша – екінші ойыншының оңтайлы стратегиясының анықталуы төмендегі функцияның максимал мәнін табу арқылы жүзеге асырылады:

$$z = \sum_{j=1}^{n} \xi_j \to \max$$

мынадай шарттар орындалуы тиіс

$$\sum_{j=1}^{n} a_{ij}\xi_j \le 1 \ (i = \overline{1,m});\ \xi_j \ge 0 \ (j = \overline{1,n});$$ мұндағы $\mathbf{\xi_j = q_j/v}$.

"mxn" ойынының шешімін табу үшін келесідей қос мағыналы есептер жұбын құру керек:

A	*B*
$\omega = \sum_{i=1}^{m} u_i \to \min$	$z = \sum_{j=1}^{n} \xi_j \to \max$
$\mu_1 = a_{11}u_1 + a_{21}u_2 + ... + a_{m1}u_m \ge 1$	$\eta_1 = a_{11}\xi_1 + a_{12}\xi_2 + ... + a_{1n}\xi_n \le 1$
.....	.....
$\mu_m = a_{1n}u_1 + a_{2n}u_2 + ... + a_{mn}u_m \ge 1$	$\eta_m = a_{m1}\xi_1 + a_{m2}\xi_2 + ... + a_{mn}\xi_n \le 1$
$u_i \ge 0,\ i = \overline{1,m}$	$\xi_j \ge 0,\ j = \overline{1,n}$

Осы есептер жұбын қос мағыналы кестеге жазайық (3.1-параграфты қараңыз):

117-кесте

		μ_1	μ_2	…	μ_n	ω
		$-\xi_1$	$-\xi_2$	…	$-\xi_n$	1
u_1	η_1	a_{11}	a_{12}	…	a_{1n}	1
u_2	η_2	a_{21}	a_{22}	…	a_{2n}	1
…	…	…	…	…	…	…
u_m	η_m	a_{m1}	a_{m2}	…	a_{mn}	1
1	z	–1	–1	…	–1	0

"mxn" ойынын шешу әдісінің алгоритмі:

1^0. Берілген төлем матрицасы бойынша ойындар теориясының негізгі теоремасына сәйкес екі ойыншы үшін есептер құрылады.

2^0. Есептер жұбы сызықтық бағдарламалау есептеріне келтіріледі.

3^0. Есептер жұбы үшін қос мағыналы кесте құрылады, В ойыншысына арналған есеп негізгі (тіке) есеп болып саналады.

4^0. Тіке есеп тіке симплекс әдісімен шешіледі (сол уақытта қос мағыналы есептің айнымалылары орын ауыстырады).

5^0. Қос мағыналы есептер жұбының оңтайлы жоспарлары анықталады.

6^0. Қос мағыналы есептер жұбының жоспарларының және оңтайлы стратегиялары мен ойын бағасының арасындағы қатынасты қолдана отырып, есептің шешімі табылады.

27-мысал. Төмендегі төлем матрицасымен берілген ойынның шешімін табу:

$$\begin{array}{c} \\ A_1 \\ A_2 \\ A_3 \end{array} \begin{array}{c} \begin{array}{ccc} B_1 & B_2 & B_3 \end{array} \\ \begin{pmatrix} -2 & -1 & 5 \\ 1 & -2 & 3 \\ 3 & 1 & -4 \end{pmatrix} \end{array}$$

Қос мағыналы есептер жұбын құрайық

A	B
$\nu \to \max$	$\nu \to \min$
$-2p_1 + p_2 + 3p_3 \geq \nu$	$-2q_1 - q_2 + 5q_3 \leq \nu$
$.- p_1 - 2p_2 + .p_3 \geq \nu$	$q_1 - 2q_2 + 3q_3 \leq \nu$
$5p_1 + 3p_2 - 4p_3 \geq \nu$	$3q_1 + q_2 - 4q_3 \leq \nu$
$\sum_i p_i = 1,\ u_i = \frac{p_i}{\nu}$	$\sum_j q_j = 1,\ \xi_j = \frac{q_j}{\nu}$

Соңғы белгілеулерді пайдалана отырып, сызықтық бағдарламалау есебіне келтірейік:

A	*B*
$\omega = \sum_{i=1}^{3} u_i \to \min$	$z = \sum_{j=1}^{3} \xi_j \to \max$
$\mu_1 = -2u_1 + u_2 + 3u_3 \geq 1$	$\eta_1 = -2\xi_1 - \xi_2 + 5\xi_3 \leq 1$
$\mu_2 = -u_1 - 2u_2 + u_3 \geq 1$	$\eta_2 = \xi_1 - 2\xi_2 + 3\xi_3 \leq 1$
$\mu_3 = 5u_1 + 3u_2 - 4u_3 \geq 1$	$\eta_m = 3\xi_1 + \xi_2 - 4\xi_3 \leq 1$
$u_i \geq 0,\ i = \overline{1,3}$	$\xi_j \geq 0,\ j = \overline{1,3}$

Симплекс кестені толтыра отырып, негізгі есепке тіке симплекс әдісін қолданайық:

118-кесте

	v_1	v_2	v_3	ω
	$-\xi_1$	$-\xi_2$	$-\xi_3$	1
$u_1\eta_1$	-2	-1	5	1
$u_2\eta_2$	1	-2	3	1
$u_3\eta_3$	3	$\boxed{1}$	-4	1
$1Z$	-1	-1	-1	0

119-кесте

	v_1	u_3	v_3	ω
	$-\xi_1$	$-\eta_2$	$-\xi_3$	1
$u_1\eta_1$	1	1	$\boxed{1}$	5
$u_2\eta_2$	7	2	-5	3
$v_2\xi_2$	3	1	-4	1
$1Z$	2	1	-5	1

120-кесте

	v_1	u_3	u_1	ω
	$-\xi_1$	$-\eta_3$	$-\eta_1$	1
$v_3\xi_1$	1	4	1	2
$u_2\eta_2$	12	7	5	13
$v_2\xi_2$	7	5	4	9
$1Z$	7	6	5	11

Жауабы:

$\xi^* = (0;9;2)$

$u^* = (5;0;6) \qquad p^* = (\frac{5}{11};0;\frac{6}{11})$

$z^* = \omega^* = 11$

$v^* = \frac{1}{11} \quad q^* = (0;\frac{9}{11};\frac{2}{11})$

10.6. Статистикалық шешімдер теориясының есептері

Статистикалық шешімдер теориясы идеялары мен әдістері бойынша ойындар теориясына жақын болып келеді. Ойындар теориясынан айырмашылығы – белгісіздік жағдайы жанжалды сипатқа ие болмайды, ешкім ешкімге қарсы тұрмайды, бірақ белгісіздік элементі айқын көрініп тұрады. Операциямен қатар жүретін белгісіздік жағдайы қарсыластың саналы қарсылығынан емес, көбінесе біздің операция өткізілетін шарттарды жетік білмеуімізден туындайды. Мысалы, мына нәрселер алдын-ала белгілі болуы мүмкін: қандай да бір аудандағы ауа райы, белгілі бір өнім түріне сатып алушылардың сұранысы, темір жол арқылы орындалатын тасымалдау көлемі, және т.б.

Осыған ұқсас барлық жағдайларда операцияны орындау шарттары бізге саналы түрде қарсы тұрған қарсыласқа байланысты емес, объективті шындыққа байланысты, шешімдер қабылдау теориясында мұны «табиғат» деп атайды. Осыған сәйкес жағдайлар «табиғатпен ойнау» деп аталады. Статистикалық шешімдер теориясында «табиғат» қандай да бір мүдделі емес инстанция ретінде қарастырылады, оның «тәртібі» белгісіз, бірақ қандай жағдайда болмасын, ол біздің жоспарларға саналы түрде қарсы тұрмайды.

Мынадай жағдайды қарастырайық. А ойыншысының m стратегиясы бар болсын: A_1, A_2, ..., A_m; жағдайға келетін болсақ, оның n ұсынысы бар делік $П_1$, $П_2$, …, $П_n$ – оларды «табиғат стратегиялары» ретінде қарастырайық. Әрбір A_i, $П_j$ стратегиялар жұбындағы a_{ij} ұтысы төлем матрицасымен берілген:

$П_j$ / A_i	$П_1$	$П_2$	...	$П_n$
A_1	a_{11}	a_{12}	…	a_{1n}
A_2	a_{21}	a_{22}	…	a_{2n}
…	…	…	…	…
A_m	a_{m1}	a_{m2}	…	a_{mn}

Басқа стратегиялармен салыстырғанда қолайлырақ (тиімдірек) болатындай етіп А ойыншысының (таза немесе аралас) стратегиясын таңдау керек.

Бірқатар жағдайларда R тәуекелдер матрицасын қарастырады, оның r_{ij} элементі А ойыншысының табиғат жағдайын біле тұрып A_i стратегиясын таңдағанда алатын ұтысы мен табиғат жағдайын білмей тұрып таңдағанда алатын ұтысы арасындағы айырым болып табылады:

$$r_{ij} = \beta_j - a_{ij}, \quad \beta_j = \max_i a_{ij}.$$

Шешімді таңдауға арналған бірнеше критерий бар.

Вальд максиминдік критерийі

Бұл критерийг бойынша, оңтайлы стратегия ретінде А ойыншысының минималды ұтысы максималды болып табылатын стратегиясы таңдалады, яғни кез келген жағдайда максиминнен кем болмайтын ұтысқа кепілдік беретін стратегия таңдалады:

$$\omega = \max_i \min_j a_{ij}$$

Егер осы критерийді басшылыққа алатын болсақ, әрқашан ең нашар шарттарға бейімделу керек және ең нашар шарттарда ұтыс максимал болатындай стратегияны таңдау керек. Табиғатпен ойнау кезінде осындай критерийді пайдалана отырып, біз осы бір жақсыз әрі мүдделі емес инстанцияның орнына белсенді және арам ниетті қарсыласты қоямыз. Мұндай тәсіл жағдайды бағалауда тым пессимистік – «әрқашан нашар жағдайға дайын болу керек!» – деген көзқарастан туындауы мүмкін, алайда қолдануға болатын тәсілдердің бірі ретінде қарастыруға болады.

Сэвидж тәуекелінің минимаксты критерийі.

Бұл критерий белгісіздік жағдайында ең қолайсыз жағдайда (тәуекел максимал болған жағдайда) тәуекелінің шамасы ең төмен мәнге ие болатын стратегияны таңдауды ұсынады:

$$S = \min_i \max_j r_{ij}$$

Бұл критерийдің мәні мынада: шешім қабылдау кезінде қалай болса да тәуекелден құтылу.

Сэвидж критерийі, Вальд критерийі тәрізді – тым пессимист критерий болып табылады, бірақ мұндағы пессимизм басқаша түсіндіріледі: ең нашар ұтыс болып минимал ұтыс емес, осы жағдайда алуға болатын ұтыспен салыстырғандағы ұтыстың максимал жоғалтуын айтады (максимал тәуекел).

Гурвиц пессимизм-оптимизм критерийі.

Бұл критерий белгісіздік жағдайында шешімді таңдау кезінде тым пессимизмді де (әрқашан нашар жағдайға дайын бол!), және тым жеңілтек оптимизмді де (бәрі де жақсы болады!) басшылыққа алмауды ұсынады. Гурвиц критерийі келесі түрге ие:

$$\mathrm{H} = \max_i \{\alpha \min_j a_{ij} + (1-\alpha) \max_i a_{ij}\},$$

мұндағы α – нөл мен бірдің арасында таңдалатын коэффициент.

α = 1 болғанда, Гурвиц критерийі пессимист Вальд критерийіне айналады, ал α = 0 болғанда, «тым оптимист» критерийге айналады, ол ең жақсы жағдайда ұтыс максимал болатын стратегияны ұсынады. $0 < \alpha < 1$ болғанда, шектен тыс пессимизм мен шектен тыс оптимизмнің арасындағы орташа мән алынады (коэффициент зерттеушінің «пессимизм өлшемін» көрсетеді). Бұл коэффициент субъективтік пайымдаулар бойынша таңдалады – жағдай қауіпті болған сайын, біз одан сайын «сақтанғымыз» келеді, сосын α 1-ге жақындай түседі.

28-мысал. 4*4 табиғат ойыны қарастырылады, ондағы ойыншының төрт стратегиясы бар: $А_1$, $А_2$, $А_3$, $А_4$ және төрт шарт бар (табиғаттың күйлері): $П_1$, $П_2$, $П_3$, $П_4$. Ұтыстар матрицасы келесі кестемен берілген:

A_i \ $П_j$	$П_1$	$П_2$	$П_3$	$П_4$
A_1	5	2	8	4
A_2	2	3	4	12
A_3	8	5	3	10
A_4	1	4	2	8

Вальд, Сэвидж және Гурвиц критерийлерін пайдалана отырып оңтайлы шешімді (стратегияны) табу керек, мұндағы $\alpha = 0{,}6$.

Шешуі.

1. Вальд критерийі.

Матрицаның әрбір жолында ең кіші ұтысты аламыз.

A_i \ $П_j$	$П_1$	$П_2$	$П_3$	$П_4$	α_i
A_1	5	2	8	4	2
A_2	2	3	4	12	2
A_3	8	5	3	10	3*
A_4	1	4	2	8	1

α_i шамаларының арасындағы максималдысы (жұлдызшамен белгіленген) 3-ке тең, демек, Вальд критерийі бойынша A_3 стратегиясы оңтайлы стратегия болып табылады.

2. Сэвидж критерийі.

Тәуекелдер матрицасын құрамыз да, оның оң жақтағы қосымша бағанына әрбір жолдағы γ_i тәуекелін жазамыз.

105-кесте

A_i \ $П_j$	$П_1$	$П_2$	$П_3$	$П_4$	γ_i.
A_1	3	3	0	8	8
A_2	6	2	4	0	6
A_3	0	0	5	2	5*
A_4	7	1	6	4	7

γ_i мәндерінің арасындағы минималдысы 5 болып табылады (жұлдызшамен белгіленген); демек, Сэвидж критерийі бойынша A_3 стратегиясы оңтайлы стратегия болып табылады.

3. Гурвиц критерийі ($\alpha = 0{,}6$)

$$H = \max_i\{0{,}6*2+0{,}4*8;\ 0{,}6*2+0{,}4*12;\ 0{,}6*3+0{,}4*10;\ 0{,}6*1+0{,}4*8\} =$$

$$= \max\{4{,}4;\ 6{,}0;\ 5{,}8;\ 3{,}8\} = 6{,}0.$$

Осылайша, A_2 – оңтайлы стратегия.

Сұрақтар мен тапсырмалар

1. Қандай жағдайларда шешім таза стратегияларда анықталады?
2. Ойынның төмен бағасын қамтамасыз ететін А ойыншысының стратегиясы қалай аталады?
3. Статистикалық шешімдерді таңдау критерийлерін атаңыз.
4. Ойындар теориясының негізгі ұғымдарына анықтама беріңіз: жүріс, стратегия, жеке жүріс, кездейсоқ жүріс, ойынның жоғарғы және төменгі бағасы.
5. Ойынның жоғары бағасын қамтамасыз ететін В ойыншысының стратегиясы қалай аталады?
6. 2 × 2, 2 × n, m × 2 өлшемді ойындық есептерді шешу әдістерін атаңыз.
7. Қандай жағдайларда ойындар теориясының есептері сызықтық бағдарламалау есептеріне келтіріледі?
8. Ойындар теориясының негізгі теоремасын тұжырымдаңыз.
9. Статистикалық шешімдерді таңдау критерийлерін келтіріңіз.
10. Келесідей төлем матрицасымен берілген ойындардың шешімін табыңыз:

$$\begin{pmatrix} 8 & 5 & 3 & 6 & 7 \\ 4 & 7 & 9 & 5 & 8 \end{pmatrix}.$$

11. Келесідей төлем матрицасымен берілген ойындардың шешімін табыңыз:

$$\begin{pmatrix} 6 & 4 \\ 5 & 3 \\ 3 & 6 \\ 1 & 8 \\ 2 & 5 \end{pmatrix}.$$

12. Келесідей төлем матрицасымен берілген ойындардың шешімін табыңыз:

$$\begin{pmatrix} 6 & 2 & 5 \\ 4 & 3 & 7 \\ 5 & 5 & 6 \end{pmatrix}.$$

11. Келесі ойын есебін шешу керек:

$$A = \begin{pmatrix} 0{,}4 & 0{,}2 \\ 0{,}1 & 0{,}5 \end{pmatrix}$$

12. Келесі ойын есебін шешу керек:

$$A = \begin{pmatrix} 4 & 7 & 4 & 2 \\ 9 & 0 & 7 & 8 \end{pmatrix}.$$

13. Келесі ойын есебін шешу керек:

$$A = \begin{pmatrix} 2 & 7 \\ 3 & 5 \\ 11 & 2 \end{pmatrix}.$$

14. Келесі ойын есебін шешу керек:

$$\begin{pmatrix} 2 & 8 \\ 4 & 3 \\ 0 & 6 \\ 3 & 4 \\ 5 & 2 \end{pmatrix}.$$

15. Келесі ойын есебін шешу керек:

$$\begin{pmatrix} 2 & 8 & 8 \\ 0 & 10 & 2 \\ 8 & 6 & 7 \end{pmatrix}.$$

16. Келесі ойын есебін шешу керек:

$$A = \begin{pmatrix} 0.3 & 0.6 & 0.8 \\ 0.9 & 0.4 & 0.2 \\ 0.7 & 0.5 & 0.4 \end{pmatrix}.$$

17. Келесі ойын есебін шешу керек:

$$A = \begin{pmatrix} 8 & 6 & 4 & 7 & 7 \\ 5 & 4 & 3 & 4 & 6 \\ 4 & 3 & 2 & 3 & 4 \\ 7 & 2 & 6 & 5 & 9 \end{pmatrix}$$

18. Келесі ойын есебін шешу керек:

$$A = \begin{pmatrix} 4 & 5 & 8 & 7 & 9 \\ 3 & 4 & 6 & 5 & 6 \\ 7 & 6 & 10 & 8 & 11 \\ 2 & 5 & 4 & 7 & 3 \end{pmatrix}$$

СІЛТЕУІШ

Есептеу алгоритмі, 19,26
- - тіке симплекс әдісі, 57
- - қос мағыналы симплекс әдісі, 87
- - потенциалдар әдісі, 100
- - венгр әдісі, 115
- - Гоморидің кесіп түсіру әдісі, 132
- - Ленд-Дойг әдісі, 136
- - Литтл, 200
- - Р.Беллман әдісі, 206
- - Форд, 178
- - Форд-Фалкерсон әдісі, 180
- - Лагранж көбейткіштерінің әдісі, 219
- - алтын қиық әдісі, 220
- - циклдық әрбір координата бойынша құлдилау әдісі, 221
- - барынша тез құлдилау әдісі, 224
- - Ньютон-Рабсон әдісі, 228
- - Зойтендейктің мүмкін бағыттар әдісі, 214
- - Франк-Вульф әдісі, 231
- - түйіндес градиент әдісі,229
- - екілік сипаттау әдісі, 241
- - 2х2 ойынының графикалық әдісі, 242
- - жиындалған ойындардың графикалық әдісі, 244
- - СБЕ-ні ОТ есебіне келтіру әдісі,246
Ациклдық жоспар, 102
Гамильтон контуры, 194
СБЕ геометриялық мағынасы, 50
Динамикалық бағдарламалау, 148
Тіректі жоспар (шешім), 29, 60, 88
Қос мағыналы есептер, 75
Қос мағыналы кестелер, 76
Теореманың қос мағыналылығы, 82, 218
Жордан шығарулары, 43
- - түрлендірілген, 58
Диета туралы есеп, 30
- өнімді оңтайлы өндіру туралы, 31
- транспорттық, 32

- тағайындаулар туралы, 33
- таңдау, 126
- материалдарды оңтайлы пішу туралы, 34
- шығындарды минималдау туралы,34
- егіс алқаптарын үлестіру туралы,35
- үлестірмелі,103
- бульдік айнымалылары бар,104
- бүтін сандық бағдарламалау,126
- жартылай бүтін сандық бағдарламалау,128
- ранец туралы, 162
- дискреттік бағдарламалау, 126
- комбинаторлық типті, 127
- ресурстарды үлестіру,154
- қорларды басқару, 158
- бір өлшемді ранец туралы, 162
- ұшақты жүктеу туралы, 33
- ең қысқа жол туралы,175, 178
- минимал бағаның максимал ағыны туралы, 177
- пошташы, 104
- коммивояжер,106
- гамильтон контуры туралы,194
- орналастыру,156
- бейсызықтық бағдарламалау, 212
Операцияларды зерттеу, 20
- - мақсаты, 20
- - негізгі кезеңдері,20
- - негізгі принциптері, 27
Итерация,
Қайнар көзі, 158
Компонент, 16
Вальд критерийі, 253
- Сэвидж, 253
- Гурвиц, 253
Модель, 17
- математикалық, 19, 24, 36, 57, 128
- сандық, 6
- логикалық, 6

- машиналық, 6
- цифрлық, 6
Бірыңғай шығарып тастау әдісі, 43
- Гаусс, 48
- тіке симплекс, 57
- қос мағыналы симплекс, 87
- потенциалдар, 100
- венгрлік, 115
- Гоморидің кесіп түсіру, 132
- Ленд-Дойг, 136
- Р.Беллман, 206
- Форда-Фалкерсон, 180
- Лагранж көбейткіштері, 219
- алтын қиық, 220
- циклдық әрбір координата бойынша құлдилау, 221
- барынша тез құлдилау, 224
- Ньютон-Рабсон, 228
- мүмкін бағыттар,233
- Франк-Вульф, 231
- түйіндес градиенттер, 229
- екілік сипаттау, 241
- 2х2 ойынының графикалық әдісі,242
- жиындалған ойындардың графикалық әдісі, 244
- СБЕ-ні ОТ есебіне келтіру,246
Объект, 15
Операция, 20
Оңтайлы жоспар (шешім), 28
Ағын, 177
Бағдарламалау, 10
- математикалық, 10
- сызықтық, 25
- бейсызықтық, 25
- дискреттік, 25
- бүтін сандық, 25
- динамикалық, 25
- тораптық, 25
- стохастикалық, 25

- детерминделген, 29
Жүйе көрінісі құрылымдық, 14
- - функционалдық,14
- - макроскопиялық, 14
- - микроскопиялық, 14
- - иерархиялық,14
- - процесуалдық, 14
Айырым, 159
Торап, 176
Жүйе, 15
- қарапайым, 14
- күрделі,16
Сток, 177
Тізбек, 177
Цикл, 177
Жүйе элементі, 16

БИБЛИОГРАФИЯЛЫҚ ТІЗІМ

1. Вентцель Е.С. Исследование операций. – М.: Советское радио, 1972, 1979, 1980. – 240 с.

2. Таха Х. Введение в ИСО. Книга 1. – М.: Мир, 1985, 2000. – 245с.

3. Таха Х. Введение в ИСО. Книга 2. – М.: Мир, 1985, 2000. – 244 с.

4. Зуховицкий С.И., Авдеева Л. И. Линейное и выпуклое программирование. – М.: Наука, 1967, 2000. – 324 с.

5. Форд Л., Фалкерсон Д. Потоки в сетях. – М.: Мир, 1966.– 400 с.

6. Хедли Дж. Нелинейное и динамическое программирование. – М.: Мир, 1967. – 327 с.

7. Химмельблау Д. Прикладное нелинейное программирование. – М.: Мир, 1972. – 247 с.

8. Канторович Л.В., Горстко А.Б. Математическое оптимальное программирование в экономике. – М.: Знание, 1968. – 257 с.

9. Майника Э. Алгоритмы оптимизации на сетях и графах. – М.: Мир,1971. – 325 с.

10. Исследование операций в экономике / Под ред. проф. Крамера Н.Ш. – М.: Юнити, 1999. – 267 с.

11. Акоф Р., Сасиени М. Основы исследования операций. – М.: Мир, 1971. – 286 с.

12. Беллман Р. Динамическое программирование. – М.: ИЛ, 1960. – 300 с.

13. Беллман Р., Дрейфус С. Прикладные задачи динамического программирования. – М.: Наука, 1965. –320 с.

14. Вагнер Г. Основы исследования операций. Т. 1-3. – М.: Мир, 1973, 287 с.

15. Гольштейн Е.Г., Юдин Д.Б. Задачи линейного программирования транспортного типа. – М.: Мир, 1969, 290 с.

16. Ху Т. Целочисленное программирование и потоки в сетях. – М.: Мир, 1974. – 323 с.

17. Корбут А.А., Финкельштейн Ю.Ю. Дискретное программирование. – М.: Наука, 1969. – 320 с.

18. Воробьев Н.Н. Теория игр. Лекции для экономистов-кибернетиков. - М.: Изд-во Ленингр. ун-та, 1974. – 250 с.

19. Вислaвский М.Н. Линейная алгебра и линейное программирование. Минск: Вышэйшая школа, 1966. – 250 с.

20. Гасс С. Линейное программирование (методы и приложения). – М.: Физматгиз, 1961. – 270 с.

21. Грень Е. Статистические игры и их применение. – М.: Статистика, 1975. – 252 с.

22. Данскин Дж.М. Теория максимина. – М.: Советское радио, 1970. – 320 с.

23. Демьянов В.Ф., Малоземов В.Н. Введение в минимакс. – М.: Наука, 1972. – 254 с.

24. Карманов В.Г. Математическое программирование. – М.: Наука, 1975. – 267 с.

25. Кофман А., Дебазей Г. Сетевые методы планирования и их применение. – М.: Прогресс, 1968. – 248 с.

26. Мак – Кинси Дж. Введение в теорию игр. – М.: Физматгиз, 1960. – 340 с.

27. Понтрягин Л.С., Болтянский В.Г., Гамкрелидзе Р.В., Мищенко Е.Ф. Математическая теория оптимальных процессов. – М.: Физматгиз, 1961. – 245с.

28. Калихман И.Л. Сборник задач по математическому программированию. – М.: Высшая школа, 1975. – 270 с.

29. Калихман И.Л., Войтенко М.А. Динамическое программирование в упражнениях и задачах. – М.: Высшая школа, 1973. – 240 с.

30. Гермейер Ю.Б. и др. Задачи по исследованию операций. Учебное пособие. – М.: Изд-во Моск. Ун-та, 1979. – 167 с.

31. Акулич И.Л. Математическое программирование в примерах и задачах. – М.: Высшая школа, 1986. – 320 с.

32. Кузнецов А.В., Холод Н.И., Костевич Л.С. Руководство к решению задач по математическому программированию. – Минск: Вышэйшая школа, 1978. – 370 с.

33. Балғабаева Л. Ш. Сызықтық және бейсызықтық бағдарламау. Тәжірибелік сабақтарға арналған әдістемелік нұсқаулар. – Алматы: ҚазҰТУ, 1997. – 19 б.

34. Балғабаева Л.Ш. Бүтінсандық бағдарламау. Транспорт есебі. Тәжірибелік сабақтарға арналған әдістемелік нұсқаулар. – Алматы: ҚазҰТУ, 1997. – 19 б.

35. Балгабаева Л.Ш., Мураталиева Е.П. Исследование операций. Методические указания к практическим занятиям и выполнению курсовой работы. – Алматы: КазНТУ, 1998. – 19 с.

36. Балғабаева Л.Ш., Мураталиева Е.П. Автомат-тандырылған басқарудың модельдері мен әдістері. Тәжірибелік сабақтарға және курстық жұмыс орындау үшін арналған әдістемелік нұсқаулар. –Алматы: ҚазҰТУ, 2003. – 54 б.

37. Балгабаева Л.Ш., Ким Е.Р. Сборник задач. Часть 1, 2, 3. Методические указания к практическим занятиям по дисциплинам «Основы исследования операций» и «Теория принятия решений». – Алматы: КазНТУ, 2005. – 32 с.

38. Балгабаева Л.Ш., Ким Е.Р. Линейное и дискретное программирование. Методические указания для выполнения контрольных работ по дисциплинам «Основы исследования операций» и «Теория принятия решений». – Алматы: КазНТУ, 2005. – 36 б.

39. Балгабаева Л.Ш., Ким Е.Р. Динамическое, сетевое и нелинейное программирование. Методические указания для выполнения контрольных работ по дисциплинам «Основы исследования операций» и «Теория принятия решений». – Алматы: КазНТУ, 2005. – 35 б.

40. Балгабаева Л.Ш., Ким Е.Р. Методы оптимизации и принятия решений. Методические указания к выполнению курсовых работ по дисциплинам «Основы исследования операций» и «Теория принятия решений». – Алматы: КазНТУ, 2005. – 18 б.

41. Балғабаева Л.Ш. Операцияларды зерттеу негіздері. Оқу құралы. – Алматы: ҚазҰТУ, 2008. – 123 б.

42. Балғабаева Л.Ш., Шанлаякова А.С. Басқарудың модельдері мен әдістері. Оқу құралы. Алматы: ҚазҰТУ, 2011. – 178 б.

МАЗМҰНЫ

	Алғы сөз ……...	3
	Кіріспе ……	5
1	ЖҮЙЕЛІК ТАЛДАУ ЖӘНЕ ОПЕРАЦИЯЛАРДЫ ЗЕРТТЕУ ӘДІСТЕМЕСІ	8
1.1	Математикалық бағдарламалау. Даму тарихы ……..	8
1.2	Жүйелік талдау және оңтайландыру ……..	11
1.3	Операцияларды зерттеудің негізгі ұғымдары мен анықтамалары ……...	20
1.4	Операцияларды зерттеудің негізгі кезеңдері …….	20
1.5	Операцияларды зерттеудің негізгі принциптері	27
1.6	Оңтайландыру әдістерінің жалпы сұлбасы ……	28
1.7	Операцияларды зерттеу есептерінің типтік кластары.....	30
1.8	Сызықтық бағдарламалаудың негізгі есебін жазудың әр түрлі формалары	36
	Сұрақтар мен тапсырмалар …….	38
2	СЫЗЫҚТЫҚ БАҒДАРЛАМАЛАУ ……	43
2.1	Бірыңғай шығарып тастау әдісі ……..	43
2.1.1	Сызықтық теңдеулер жүйесін шешу ……	43
2.1.2	Кері матрицаны есептеу ……..	45
2.1.3	Нөлдік жолдарды шығарып тастау …….	46
2.1.4	Гаусс әдісі ……..	48
2.2	Сызықтық бағдарламалау есебінің геометриялық мағынасы	50
2.3	Тіке симплекс әдісі ……...	57
2.4	Шешуші элементті таңдау ережелерін негіздеу	63
2.4.1	Тіректі шешімді табу кезіндегі шешуші элементті таңдау ережесін негіздеу	63
2.4.2	Оңтайлы шешімді табу үшін шешуші элементті таңдау ережесін негіздеу	66
2.4.3	Нөлдік жолдарды шығарып тастау кезінде шешуші элементті таңдау ережесін негіздеу	67
	Сұрақтар мен тапсырмалар …….	68
3	СЫЗЫҚТЫҚ БАҒДАРЛАМАЛАУДАҒЫ ҚОС МАҒЫНАЛЫЛЫҚ ……...	75

3.1	Қос мағыналы есептер …….	75
3.2	Қос мағыналы кестелер ……	76
3.3	Қос мағыналылық теоремалары ……..	82
3.4	Қос мағыналы есептер жұбының экономикалық түсіндірмесі	86
3.5	Қос мағыналы симплекс әдісі …….....	87
	Сұрақтар мен тапсырмалар …….	91
4	ТРАНСПОРТ ЕСЕБІ ……	97
4.1	Транспорт есебінің жалпы қойылымы ……...	97
4.2	Потенциалдар әдісі …….....	100
4.3	Венгр әдісі ……..……	115
	Сұрақтар мен тапсырмалар …….	120
5	БҮТІН САНДЫҚ БАҒДАРЛАМАЛАУ …….	126
5.1	Бүтін сандық бағдарламалау есебінің графикалық түсіндірілуі …….....	130
5.2	Гоморидің кесіп түсіру әдісі ……	132
5.3	Ленд-Дойг әдісі ……	136
	Сұрақтар мен тапсырмалар …….	141
6	ДИНАМИКАЛЫҚ БАҒДАРЛАМАЛАУ ……...	148
6.1	Динамикалық бағдарламалау туралы жалпы мәліметтер және негізгі ұғымдар	148
6.2	Динамикалық бағдарламалау әдісінің есептері	154
6.3	Динамикалық бағдарламалау есебінің жалпы қойылымы	155
6.4	Оңтайлы басқаруды кезең бойынша құру принципі	155
6.5	Динамикалық бағдарламалаудың есептеу сұлбасы	157
6.6	Динамикалық бағдарламалаудың математикалық моделін құру ережелері	157
6.7	Бір өлшемді ранец туралы есеп …….....	162
	Сұрақтар мен тапсырмалар …….	170
7	ТОРАПТЫҚ ЖОСПАРЛАУ ЖӘНЕ БАСҚАРУ …...	173
7.1	Жалпы мәліметтер мен ұғымдар …….	173
7.2	Тораптық жоспарлау есептері …….	175
7.3	Негізгі ұғымдар мен анықтамалар ……..	176
7.4	Ең қысқа жол туралы есеп ……...	178
7.5	Максималды ағын және минималды баға туралы есеп. Форд-Фалкерсон әдісі ……	180

7.6	Тораптық графиктердің жоспарлы параметрлерін есептеу ……..	184
	Сұрақтар мен тапсырмалар …….	191
8	КОММИВОЯЖЕР ЕСЕБІ ……	194
8.1	Коммивояжер есебі тағайындаулар туралы есептің бір түрі ретінде	199
8.2	Р. Беллман алгоритмін коммивояжер есебіне қолдану	206
	Сұрақтар мен тапсырмалар …….	209
9	БЕЙСЫЗЫҚТЫҚ БАҒДАРЛАМАЛАУ …….	212
9.1	Негізгі ұғымдар. Бейсызықтық бағдарламалау әдістерінің жіктелуі	212
9.2	Бейсызықтық бағдарламалау есептерінің қос мағыналылығы	216
9.3	Қос мағыналылық теоремалары ……..	218
9.4	Жалпы бейсызықтық бағдарламалау есебінің дербес жағдайы	218
9.5	Шартсыз оңтайландырудың бейсызықтық бағдарламалау есептері	220
9.5.1	Алтын қиық әдісі ……..	220
9.5.2	Циклдық әрбір координата бойынша құлдилау әдісі	221
9.6	Туындыларды қолданатын көп өлшемді шартсыз оңтайландыру есептері	224
9.6.1	Барынша тез құлдилау әдісі	224
9.6.2	Түйіндес градиент әдісі ……...	227
9.6.3	Ньютон-Рабсон әдісі (екінші туындылар әдісі)	228
9.7	Шартты оңтайландырудың көп өлшемді бейсызықтық есептері ……..	231
9.7.1	Франк-Вульф әдісі ……	231
9.7.2	Зойтендейктің мүмкін бағыттар әдісі …….	233
	Сұрақтар мен тапсырмалар …….	235
10	ОЙЫНДАР ЖӘНЕ СТАТИСТИКАЛЫҚ ШЕШІМДЕР ТЕОРИЯСЫ	238
10.1	Негізгі ұғымдар мен анықтамалар	238
10.2	Екілік сипаттау әдісі (алгебралық әдіс)	241
10.3	2×2 ойынын шешудің графикалық әдісі ……..	242

10.4 Жиындалған ойындарға графикалық әдісті қолдану 244
10.5 Ойындар теориясының есептерін сызықтық бағдарламалау есептеріне келтіру 246
10.6 Статистикалық шешімдер теориясының есептері … 251
Сұрақтар мен тапсырмалар ……. 255
Сілтеуіш 258
Библиографиялық тізім 262

Оқулық басылым

Ләззат Шайханқызы Балғабаева

ОПЕРАЦИЯЛАРДЫ ЗЕРТТЕУ НЕГІЗДЕРІ

Оқулық

ОБО РБ бастығы *З. Ғұбайдулина*
Редакторы *А. Бейсебаева*
Компьютерде беттеген *Л. Касжанова*

Басуға қол қойылды 28.04.2016 ж.
Таралымы 300 дана. Пішімі 60х84х 1/16. № 1 баспаханалық қағаз.
Көлемі 16,8 есепті-баспа табақ. Шартты баспа табақ. 15,6.
Тапсырыс № 741. Бағасы келісімді.

Қ. И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу
университетінің басылымы.
Оқу-баспа орталығы,
Алматы қ., Сәтбаев көшесі, 22

ISBN 978-601-228-940-4