МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования   
«Национальный исследовательский   
Нижегородский государственный университет им. Н.И. Лобачевского»

**(ННГУ)**

**РЕКОМЕНДАЦИИ ПО ОФОРМЛЕНИЮ ОТЧЕТНЫХ   
И КВАЛИФИКАЦИОННЫХ РАБОТ**

*Учебно-методические рекомендации*

*Для студентов, обучающихся   
в институте информационных технологий,  
 математики и механики*

Нижний Новгород  
 2019

Введение

Планирование и управление комплексом работ по проекту представляет собой сложную и, как правило, противоречивую задачу. Оценка временных и стоимостных параметров функционирования системы, осуществляемая в рамках этой задачи, производится различными методами. Среди существующих большое значение имеет метод сетевого планирования.

Сетевое планирование — метод анализа сроков (ранних и поздних) начала и окончания нереализованных частей проекта, позволяет увязать выполнение различных работ и процессов во времени, получив прогноз общей продолжительности реализации всего проекта.

Методы сетевого планирования могут широко и успешно применяются для оптимизации планирования и управления сложными разветвленными комплексами работ, которые требуют участия большого числа исполнителей и затрат ограниченных ресурсов.

Первый этап широкого использования сетевого планирования был связан с появлением диаграмм Ганта, которые появились в начале двадцатого века. Диаграмма Ганга это удобный инструмент для организации, планирования и управления ходом выполнения самых разнообразных процессов.

Второй этап. Методики сетевого планирования были разработаны в конце 50-х годов в США. В 1956 г. М. Уолкер из фирмы "Дюпон", исследуя возможности более эффективного использования принадлежащей фирме вычислительной машины Univac, объединил свои усилия с Д. Келли из группы планирования капитального строительства фирмы "Ремингтон Рэнд". Они попытались использовать ЭВМ для составления планов-графиков крупных комплексов работ по модернизации заводов фирмы "Дюпон". В результате был создан рациональный и простой метод описания проекта с использованием ЭВМ. Первоначально он был назван методом Уолкера-Келли, а позже получил название метода критического пути — МКП (или CPM —Critical Path Method).

Параллельно и независимо в военно-морских силах США был создан метод анализа и оценки программ PERT (Program Evaluation andReview Technique). Данный метод был разработан корпорацией "Локхид" и консалтинговой фирмой "Буз, Аллен энд Гамильтон" для реализации проекта разработки ракетной системы "Поларис", который объединял около 3800 основных подрядчиков и состоящего из 60 тыс. операций. Использование метода PERT позволило руководству программы точно знать, что требуется делать в каждый момент времени и кто именно должен это делать, а также вероятность своевременного завершения отдельных операций. Проект удалось завершить на два года раньше запланированного срока благодаря успешному руководству программы.

Третий этап связан как с продолжавшимся в конце двадцатого века усовершенствованием прежних методов управления проектами, так и с появлением новых, но на более качественном уровне - с применением современного программного обеспечения и персональных компьютеров. Сначала разработка программного обеспечения велась крупными компаниями с целью поддержки собственных проектов, но вскоре первые системы управления проектами появились и на рынке программного обеспечения. Системы, стоявшие у истоков планирования, разрабатывались для мощных больших компьютеров и сетей мини-ЭВМ.

Следует отметить, что главной целью сетевого планирования является сокращение до минимума продолжительности проекта, таким образом, использование сетевых моделей обусловлено необходимостью грамотного управления крупными проектами и предприятиями, научными исследованиями, конструкторской и технологической подготовкой производства, новых видов изделий, строительством и реконструкцией, капитальным ремонтом основных фондов и т.п.

С помощью сетевой модели руководитель работ или операции может системно и масштабно представлять весь ход работ или оперативных мероприятий, Управлять процессом их осуществления, а также маневрировать ресурсами.

Оглавление

[Глава 1. 5](#_Toc1994570)

[Задача расчета временных характеристик 5](#_Toc1994571)

[Построение математической модели распределения ресурсов 12](#_Toc1994572)

[Глава 2 26](#_Toc1994573)

[Методы решения задач оперативного управления для автономной системы сетевой структуры 26](#_Toc1994574)

[Заключение 27](#_Toc1994575)

[Список литературы 28](#_Toc1994576)

# Глава 1.

Задача расчета временных характеристик

Склады представляют собой разнообразные помещения, где содержатся товары, и различные устройства, специально предназначенные для их приемки, размещения и хранения. Сегодня склад – это хорошо отрегулированная многоуровневая организация, объединенная в единый технологический процесс с автоматизированными системами по учету складируемых запасов, начиная от их приемки и заканчивая отпуском конечному потребителю.

Таким образом, современный склад является сложной структурой, как с технической, так и с управленческой стороны. Ускоряющиеся темпы научно-технического прогресса вносят решительные изменения в структуру логистического процесса управления запасами. Это проявляется в том, что ежедневно в наш обиход входят новые, более совершенные системы движения материальных потоков.

Широкое распространение и энергичное внедрение современных комплексных автоматизированных систем управления складом, основанных на новейших средствах получения и обработки информации в режиме реального времени обуславливается необходимостью снижения временных и трудовых затрат.

Общая математическая модель задачи распределения ограниченных ресурсов включает в себя задание исходных параметров, варьируемых параметров и ограничений математической модели.

Исходные параметры:

j – номер работы, j∈J, где J - множество всех работ для всех изделий. Работы имеют сквозную нумерацию по всем изделиям.

i – номер ресурса, i∈I, где I - множество всех ресурсов производственной системы. Ресурсы представляются складируемыми (множество I⁰).

T – множество тактов периода планирования, включает такты tₙ, tₙ₊₁, … , tₖ - натуральные числа, tₙ≥I.

Jᵐ – множество работ, у которых заданы начальные сроки, Jᵐ ⊆ J.

Jᶠ – множество работ, обладающих директивными сроками окончания работ, Jᶠ ⊆ J.

tmj  - начальный срок работы J , tmj ≥ tm, tm j∈ T, j ∈ J.

tfj  - директивный срок работы j, j ∈ J.

r i,j – ресурсоемкость j-ой работы по ш –ому ресурсу, i ∈ I, j ∈ J, r i,j ≥ , t ∈ T, j ∈ J.

Qi,t – количество i-ого ресурса поступившего в такт t для выполнения работ, i ∈ I, t ∈ T.

K(j) – множество работ непосредственно предшествующих работе j , K(j) ⊆ J, j ∈ j.

Варьируемые параметры:

xj – номер такта начала работы j, xj ∈ T, j ∈ J.

yj – номер такта начала работы j, yj ∈ T, j ∈ J.

Zi,j,t  - интенсивность потребления j-ой работой i-ого ресурса в такт t, zi,j,t  ≥ 0, i ∈ I, t ∈ T, j ∈ J.

Возможные условия функционирования производственной системы разобьем на три класса:

1. Технологические условия (обусловлены заданной технологией изготовления совокупности сложных изделий).
2. Ресурсные условия (связанные с наличием ресурсов по тактам периода планирования).
3. Организационные условия (определяются заданием директивных и начальных сроков изготовления).

В свою очередь три класса возможных условий разобьем на две группы:

1. «Жесткие» условия – их нарушение недопустимо.
2. «Мягкие» условия – можно допустить их нарушение, но это приведет к наложению штрафных санкций на производственную систему.

На рисунке отражена связь между типами и классами возможных условий.

Рис. Классификация возможных условий математической модели распределения ресурсов складируемого и не складируемого типов.

Рис. Связь между типами и классами возможных ограничений.

Расчет сетевой модели начинают с временных параметров событий, которые вписывают непосредственно в вершины сетевого графика (рис.1):

https://pandia.ru/text/78/183/images/image001_180.gif – ранний срок наступления события i, минимально необходимый для выполнения всех работ, которые предшествуют событию i;

https://pandia.ru/text/78/183/images/image002_149.gif – поздний срок наступления события i, превышение которого вызовет аналогичную задержку наступления завершающего события сети;

https://pandia.ru/text/78/183/images/image003_124.gif – резерв события i, т. е. время, на которое может быть отсрочено наступление события i без нарушения сроков завершения проекта в целом.

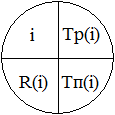


Рис.1. Отображение временных параметров событий на сетевом графике

Ранние сроки свершения событий https://pandia.ru/text/78/183/images/image001_180.gif рассчитываются от исходного (И) к завершающему (З) событию следующим образом:

1) для исходного события И https://pandia.ru/text/78/183/images/image005_95.gif;

2) для всех остальных событий Ihttps://pandia.ru/text/78/183/images/image006_87.gif,

где максимум берется по всем работам https://pandia.ru/text/78/183/images/image007_88.gif, входящим в событие i; https://pandia.ru/text/78/183/images/image008_79.gif – длительность работы (k, i) (рис.2).

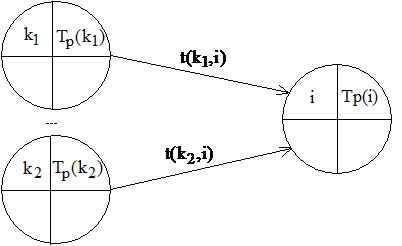


Рис.2. Расчет раннего срока https://pandia.ru/text/78/183/images/image001_180.gif свершения события i

Поздние сроки свершения событий https://pandia.ru/text/78/183/images/image002_149.gif рассчитываются от завершающего к исходному событию:

1) для завершающего события З https://pandia.ru/text/78/183/images/image010_71.gif;

2) для всех остальных событийhttps://pandia.ru/text/78/183/images/image011_66.gif,

где минимум берется по всем работам https://pandia.ru/text/78/183/images/image012_67.gif, выходящим из события i; https://pandia.ru/text/78/183/images/image013_62.gif – длительность работы (k, i) (рис.3).

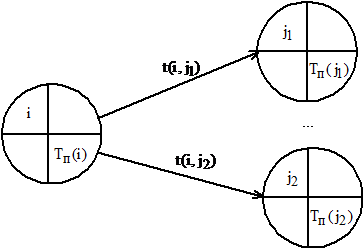


Рис.3. Расчет позднего срока https://pandia.ru/text/78/183/images/image002_149.gif свершения события i

Временные параметры работ определяются на основе ранних и поздних сроков событий:

https://pandia.ru/text/78/183/images/image015_58.gif – ранний срок начала работы;

https://pandia.ru/text/78/183/images/image016_53.gif – ранний срок окончания работы;

https://pandia.ru/text/78/183/images/image017_50.gif – поздний срок окончания работы;

https://pandia.ru/text/78/183/images/image018_49.gif – поздний срок начала работы;

https://pandia.ru/text/78/183/images/image019_45.gif – полный резерв работы показывает максимальное время, на которое можно увеличить длительность работы https://pandia.ru/text/78/183/images/image012_67.gif или отсрочить ее начало, чтобы не нарушился срок завершения проекта в целом;

https://pandia.ru/text/78/183/images/image020_47.gif – свободный резерв работы показывает максимальное время, на которое можно увеличить продолжительность работы https://pandia.ru/text/78/183/images/image012_67.gif или отсрочить ее начало, не меняя ранних сроков начала последующих работ.

Путь – это последовательность работ в сетевом графике (в частном случае это одна работа), в которой конечное событие одной работы совпадает с начальным событием следующей за ней работы. Полный путь – это путь от исходного до завершающего события. Критический путь –максимальный по продолжительности полный путь. Работы, лежащие на критическом пути, называют критическими. Критические работы имеют нулевые свободные и полные резервы. Подкритический путь – полный путь, ближайший по длительности к критическому пути.

Для проведения анализа временных параметров сетевой модели используют график привязки, который отображает взаимосвязь выполняемых работ во времени. По вертикальной оси графика привязки откладываются коды работ, по горизонтальной оси – отрезки, соответствующие длительностям работ (раннее начало и раннее окончание работ). График привязки можно построить на основе данных о продолжительности работ. При этом необходимо помнить, что работа https://pandia.ru/text/78/183/images/image021_44.gif может выполняться только после того как будут выполнены все предшествующие ей работы https://pandia.ru/text/78/183/images/image022_43.gif.

Построение математической модели распределения ресурсов

Важнейшим показателем сетевого графика являются резервы времени. Резервы времени каждого пути показывают, на сколько может быть увеличена продолжительность данного пути без ущерба для наступления завершающего события. Поскольку каждый некритический путь сетевого графика имеет свой полный резерв времени, то и каждое событие этого пути имеет свой резерв времени.

|  |  |  |
| --- | --- | --- |
| Элемент сети | Наименование параметра | Условное обозначение параметра |
| Событие i | Ранний срок свершения события | tp(i) |
|  | Поздний срок свершения события | t(i) |
|  | Резерв времени события | R(i) |
| Работа (i, j) | Продолжительность работы | t(i,j) |
|  | Ранний срок начала работы | tрн(i,j) |
|  | Ранний срок окончания работы | tpo(i,j) |
|  | Поздний срок начала работы | tпн(i,j) |
|  | Поздний срок окончания работы | tпо(i,j) |
|  | Полный резерв времени работы | Rп(i,j) |
| Путь L | Продолжительность пути | t(L) |
|  | Продолжительность критического пути | tkp |
|  | Резерв времени пути | R(L) |

Решение.

Резерв времени события показывает, на какой допустимый период времени можно задержать наступление этого события, не вызывая при этом увеличения срока выполнения комплекса работ.

Для определения резервов времени по событиям сети рассчитывают наиболее ранние tp и наиболее поздние tп сроки свершения событий. Любое событие не может наступить прежде, чем свершаться все предшествующие ему события и не будут выполнены все предшествующие работы. Поэтому ранний (или ожидаемый) срок tp(i) свершения i-ого события определяется продолжительностью максимального пути, предшествующего этому событию:

tp(i) = max(t(Lni)), где Lni – любой путь, предшествующий i-ому событию, то есть путь от исходного до i-ого события сети.

Если событие j имеет несколько предшествующих путей, а следовательно, несколько предшествующих событий i, то ранний срок свершения события j удобно находить по формуле:

tp(j) = max[tp(i) + t(i,j)]

Задержка свершения события i по отношению к своему раннему сроку не отразится на сроке свершения завершающего события (а значит, и на сроке выполнения комплекса работ) до тех пор, пока сумма срока свершения этого события и продолжительности (длины) максимального из следующих за ним путей не превысит длины критического пути. Поэтому поздний (или предельный) срок tп(i) свершения i-ого события равен: tп(i) = tkp - max(t(Lci))   
где Lci - любой путь, следующий за i-ым событием, т.е. путь от i-ого до завершающего события сети.

Если событие i имеет несколько последующих путей, а следовательно, несколько последующих событий j, то поздний срок свершения события i удобно находить по формуле:

tп(i) = min[tп(j) - t(i,j)]

Резерв времени R(i) i-ого события определяется как разность между поздним и ранним сроками его свершения: R(i) = tп(i) - tp(i)

Резерв времени события показывает, на какой допустимый период времени можно задержать наступление этого события, не вызывая при этом увеличения срока выполнения комплекса работ.

Критические события резервов времени не имеют, так как любая задержка в свершении события, лежащего на критическом пути, вызовет такую же задержку в свершении завершающего события. Таким образом, определив ранний срок наступления завершающего события сети, мы тем самым определяем длину критического пути.

При определении ранних сроков свершения событий tp(i) двигаемся по сетевому графику слева направо и используем формулы (1), (2).

Расчет сроков свершения событий.

Для i=1 (начального события), очевидно tp(1)=0.

i=2: tp(2) = tp(1) + t(1,2) = 0 + 8 = 8.

i=3: tp(3) = tp(2) + t(2,3) = 8 + 16 = 24.

i=4: tp(4) = tp(2) + t(2,4) = 8 + 16 = 24.

i=5: tp(5) = tp(3) + t(3,5) = 24 + 3 = 27.

i=6: tp(6) = tp(4) + t(4,6) = 24 + 3 = 27.

i=7: tp(7) = tp(5) + t(5,7) = 27 + 4 = 31.

i=8: tp(8) = tp(6) + t(6,8) = 27 + 8 = 35.

i=9: tp(9) = tp(7) + t(7,9) = 31 + 4 = 35.

i=10: tp(10) = tp(7) + t(7,10) = 31 + 16 = 47.

i=11: max(tp(2) + t(2,11);tp(8) + t(8,11);tp(9) + t(9,11);tp(10) + t(10,11)) = max(8 + 20;35 + 0;35 + 32;47 + 32) = 79.

Длина критического пути равна раннему сроку свершения завершающего события 11: tkp=tp(11)=79

При определении поздних сроков свершения событий tп(i) двигаемся по сети в обратном направлении, то есть справа налево и используем формулы (3), (4).

Для i=11 (завершающего события) поздний срок свершения события должен равняться его раннему сроку (иначе изменится длина критического пути): tп(11)= tр(11)=79.

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 10. Просматриваются все строчки, начинающиеся с номера 10.

i=10: tп(10) = tп(11) - t(10,11) = 79 - 32 = 47.

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 9. Просматриваются все строчки, начинающиеся с номера 9.

i=9: tп(9) = tп(11) - t(9,11) = 79 - 32 = 47.

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 8. Просматриваются все строчки, начинающиеся с номера 8.

i=8: tп(8) = tп(11) - t(8,11) = 79 - 0 = 79.

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 7. Просматриваются все строчки, начинающиеся с номера 7.

i=7: min(tп(9) - t(7,9);tп(10) - t(7,10)) = min(47 - 4;47 - 16) = 31.

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 6. Просматриваются все строчки, начинающиеся с номера 6.

i=6: tп(6) = tп(8) - t(6,8) = 79 - 8 = 71.

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 5. Просматриваются все строчки, начинающиеся с номера 5.

i=5: tп(5) = tп(7) - t(5,7) = 31 - 4 = 27.

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 4. Просматриваются все строчки, начинающиеся с номера 4.

i=4: tп(4) = tп(6) - t(4,6) = 71 - 3 = 68.

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 3. Просматриваются все строчки, начинающиеся с номера 3.

i=3: tп(3) = tп(5) - t(3,5) = 27 - 3 = 24.

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 2. Просматриваются все строчки, начинающиеся с номера 2.

i=2: min(tп(3) - t(2,3);tп(4) - t(2,4);tп(11) - t(2,11)) = min(24 - 16;68 - 16;79 - 20) = 8.

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 1. Просматриваются все строчки, начинающиеся с номера 1.

i=1: tп(1) = tп(2) - t(1,2) = 8 - 8 = 0.

Таблица 1 - Расчет резерва событий

|  |  |  |  |
| --- | --- | --- | --- |
| Номер события | Сроки свершения события: ранний tp(i) | Сроки свершения события: поздний tп(i) | Резерв времени, R(i) |
| 1 |  | 0 | 0 |
| 2 | 8 | 8 | 0 |
| 3 | 24 | 24 | 0 |
| 4 | 24 | 68 | 44 |
| 5 | 27 | 27 | 0 |
| 6 | 27 | 71 | 44 |
| 7 | 31 | 31 | 0 |
| 8 | 35 | 79 | 44 |
| 9 | 35 | 47 | 12 |
| 10 | 47 | 47 | 0 |
| 11 | 79 | 79 | 0 |

Заполнение таблицы 2

Перечень работ и их продолжительность перенесем во вторую и третью графы. При этом работы следует записывать в графу 2 последовательно: сначала начиная с номера 1, затем с номера 2 и т.д.

Во второй графе поставим число, характеризующее количество непосредственно предшествующих работ (КПР) тому событию, с которого начинается рассматриваемая работа.

Так, для работы (1,2) в графу 1 поставим число 0, т.к. на номер 1 не оканчивается ни одна работа.

Графу 4 получаем из таблицы 1 (tp(i)). Графу 7 получаем из таблицы 1 (tп(i)).

Значения в графе 5 получаются в результате суммирования граф 3 и 4.

В графе 6 позднее начало работы определяется как разность позднего окончания этих работ и их продолжительности (из значений графы 7 вычитаются данные графы 3);

Содержимое графы 8 (полный резерв времени R(ij)) равно разности граф 6 и 4 или граф 7 и 5. Если R(ij) равен нулю, то работа является критической

Полный резерв пути показывает, на сколько в сумме может быть увеличена продолжительность всех работ, принадлежащих данному пути, при условии, что срок выполнения всего комплекса работ не изменится. Образовывается, когда предшествующие работы закончатся в свой наиболее ранний срок.

Находим полный резерв RПi-j = Tпj-ti-j-Tрi:

RП(1,2) = 8-8-0 = 0

RП(2,3) = 24-16-8 = 0

RП(2,4) = 68-16-8 = 44

RП(2,11) = 79-20-8 = 51

RП(3,5) = 27-3-24 = 0

RП(4,6) = 71-3-24 = 44

RП(5,7) = 31-4-27 = 0

RП(6,8) = 79-8-27 = 44

RП(7,9) = 47-4-31 = 12

RП(7,10) = 47-16-31 = 0

RП(8,11) = 79-0-35 = 44

RП(9,11) = 79-32-35 = 12

RП(10,11) = 79-32-47 = 0

Свободный резерв времени также можно найти и по формуле RCi-j = Tпi-ti-j-Tрi:

RC(1,2) = 8-8-0 = 0

RC(2,3) = 24-16-8 = 0

RC(2,4) = 24-16-8 = 0

RC(2,11) = 79-20-8 = 51

RC(3,5) = 27-3-24 = 0

RC(4,6) = 27-3-24 = 0

RC(5,7) = 31-4-27 = 0

RC(6,8) = 35-8-27 = 0

RC(7,9) = 35-4-31 = 0

RC(7,10) = 47-16-31 = 0

RC(8,11) = 79-0-35 = 44

RC(9,11) = 79-32-35 = 12

RC(10,11) = 79-32-47 = 0

Независимый резерв времени также можно найти и по формуле RНi-j = Tрj-ti-j-Tпi:

RН(1,2) = 8-8-0 = 0

RН(2,3) = 24-16-8 = 0

RН(2,4) = 24-16-8 = 0

RН(2,11) = 79-20-8 = 51

RН(3,5) = 27-3-24 = 0

RН(4,6) = 27-3-68 = -44

RН(5,7) = 31-4-27 = 0

RН(6,8) = 35-8-71 = -44

RН(7,9) = 35-4-31 = 0

RН(7,10) = 47-16-31 = 0

RН(8,11) = 79-0-79 = 0

RН(9,11) = 79-32-47 = 0

RН(10,11) = 79-32-47 = 0

Таблица 2 - Анализ сетевой модели по времени

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Работа (i,j) | Количество предшествующих работ | Продолжительность tij | Ранние сроки: начало tijР.Н. | Ранние сроки: окончание tijР.О. | Поздние сроки: начало tijП.Н. | Поздние сроки: окончание tijП.О. | Резервы времени: полный RijП | Независимый резерв времени RijН | Частный резерв I рода, Rij1 | Частный резерв II рода, RijC |
| (1,2) | 0 | 8 | 0 | 8 | 0 | 8 | 0 | 0 | 0 | 0 |
| (2,3) | 1 | 16 | 8 | 24 | 8 | 24 | 0 | 0 | 0 | 0 |
| (2,4) | 1 | 16 | 8 | 24 | 52 | 68 | 44 | 0 | 44 | 0 |
| (2,11) | 1 | 20 | 8 | 28 | 59 | 79 | 51 | 51 | 51 | 51 |
| (3,5) | 1 | 3 | 24 | 27 | 24 | 27 | 0 | 0 | 0 | 0 |
| (4,6) | 1 | 3 | 24 | 27 | 68 | 71 | 44 | -44 | 0 | 0 |
| (5,7) | 1 | 4 | 27 | 31 | 27 | 31 | 0 | 0 | 0 | 0 |
| (6,8) | 1 | 8 | 27 | 35 | 71 | 79 | 44 | -44 | 0 | 0 |
| (7,9) | 1 | 4 | 31 | 35 | 43 | 47 | 12 | 0 | 12 | 0 |
| (7,10) | 1 | 16 | 31 | 47 | 31 | 47 | 0 | 0 | 0 | 0 |
| (8,11) | 1 | 0 | 35 | 35 | 79 | 79 | 44 | 0 | 0 | 44 |
| (9,11) | 1 | 32 | 35 | 67 | 47 | 79 | 12 | 0 | 0 | 12 |
| (10,11) | 1 | 32 | 47 | 79 | 47 | 79 | 0 | 0 | 0 | 0 |

Следует отметить, что кроме полного резерва времени работы, выделяют еще три разновидности резервов. Частный резерв времени первого вида R1 - часть полного резерва времени, на которую можно увеличить продолжительность работы, не изменив при этом позднего срока ее начального события. R1 находится по формуле:

R(i,j)= Rп(i,j) - R(i)

Частный резерв времени второго вида, или свободный резерв времени Rc работы (i,j) представляет собой часть полного резерва времени, на которую можно увеличить продолжительность работы, не изменив при этом раннего срока ее конечного события. Rc находится по формуле:

R(i,j)= Rп(i,j) - R(j)

Значение свободного резерва времени работы указывает на расположение резервов, необходимых для оптимизации.

Независимый резерв времени Rн работы (i,j) - часть полного резерва, получаемая для случая, когда все предшествующие работы заканчиваются в поздние сроки, а все последующие начинаются в ранние сроки. Rн находится по формуле:

R(i,j)= Rп(i,j)- R(i) - R(j)

Критический путь: (1,2)(2,3)(3,5)(5,7)(7,10)(10,11)

Продолжительность критического пути: 79

Анализ сетевого графика

Сложность сетевого графика оценивается коэффициентом сложности, который определяется по формуле:

Kc = npab / ncob, где Kc – коэффициент сложности сетевого графика; npab – количество работ, ед.; ncob – количество событий, ед.

Сетевые графики, имеющие коэффициент сложности от 1,0 до 1,5, являются простыми, от 1,51 до 2,0 – средней сложности, более 2,1 – сложными. Kc = 13 / 11 = 1.18

Поскольку Kc < 1.5, то сетевой график является простым.

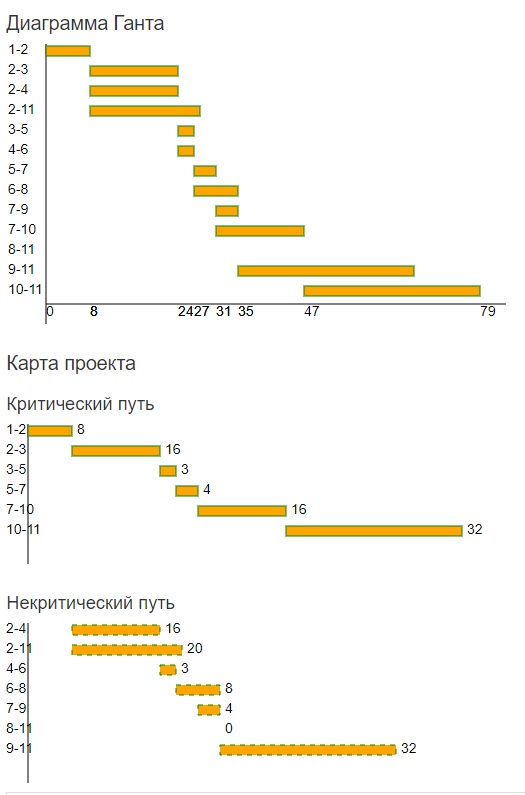
Коэффициентом напряженности КH работы Pi,j называется отношение продолжительности несовпадающих (заключенных между одними и теми же событиями) отрезков пути, одним из которых является путь максимальной продолжительности, проходящий через данную работу, а другим – критический путь:

https://chart.googleapis.com/chart?cht=tx&chl=K_%7bH%7d%20=%20\frac%7bt(Lmax)-t1_%7bkp%7d%7d%7bt_%7bkp%7d-t1_%7bkp%7d%7d где t(Lmax) – продолжительность максимального пути, проходящего через работу Pi,j, от начала до конца сетевого графика; tkp – продолжительность (длина) критического пути; t1kp – продолжительность отрезка рассматриваемого максимального пути, совпадающего с критическим путем.

Коэффициент напряженности КH работы Pi,j может изменяться в пределах от 0 (для работ, у которых отрезки максимального из путей, не совпадающие с критическим путем, состоят из фиктивных работ нулевой продолжительности) до 1 (для работ критического пути). Чем ближе к 1 коэффициент напряженности КH работы Pi,j, тем сложнее выполнить данную работу в установленные сроки. Чем ближе Кн работы Pi,j к нулю, тем большим относительным резервом обладает максимальный путь, проходящий через данную работу.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Работа | Путь | Максимальный путь, t(Lmax) | Совпадающие работы | t1kp | Расчет | КH |
| (1,2) | (1,2)(2,3)(3,5)(5,7)(7,10)(10,11) | 79 | (1,2)(2,3)(3,5)(5,7)(7,10)(10,11) | 79 | - | - |
| (2,3) | (1,2)(2,3)(3,5)(5,7)(7,10)(10,11) | 79 | (1,2)(2,3)(3,5)(5,7)(7,10)(10,11) | 79 | - | - |
| (2,4) | (1,2)(2,4)(4,6)(6,8)(8,11) | 35 | (1,2) | 8 | (35-8)/(79-8) | 0.38 |
| (2,11) | (1,2)(2,11) | 28 | (1,2) | 8 | (28-8)/(79-8) | 0.282 |
| (3,5) | (1,2)(2,3)(3,5)(5,7)(7,10)(10,11) | 79 | (1,2)(2,3)(3,5)(5,7)(7,10)(10,11) | 79 | - | - |
| (4,6) | (1,2)(2,4)(4,6)(6,8)(8,11) | 35 | (1,2) | 8 | (35-8)/(79-8) | 0.38 |
| (5,7) | (1,2)(2,3)(3,5)(5,7)(7,10)(10,11) | 79 | (1,2)(2,3)(3,5)(5,7)(7,10)(10,11) | 79 | - | - |
| (6,8) | (1,2)(2,4)(4,6)(6,8)(8,11) | 35 | (1,2) | 8 | (35-8)/(79-8) | 0.38 |
| (7,9) | (1,2)(2,3)(3,5)(5,7)(7,9)(9,11) | 67 | (1,2)(2,3)(3,5)(5,7) | 31 | (67-31)/(79-31) | 0.75 |
| (7,10) | (1,2)(2,3)(3,5)(5,7)(7,10)(10,11) | 79 | (1,2)(2,3)(3,5)(5,7)(7,10)(10,11) | 79 | - | - |
| (8,11) | (1,2)(2,4)(4,6)(6,8)(8,11) | 35 | (1,2) | 8 | (35-8)/(79-8) | 0.38 |
| (9,11) | (1,2)(2,3)(3,5)(5,7)(7,9)(9,11) | 67 | (1,2)(2,3)(3,5)(5,7) | 31 | (67-31)/(79-31) | 0.75 |
| (10,11) | (1,2)(2,3)(3,5)(5,7)(7,10)(10,11) | 79 | (1,2)(2,3)(3,5)(5,7)(7,10)(10,11) | 79 | - | - |

Вычисленные коэффициенты напряженности позволяют дополнительно классифицировать работы по зонам. В зависимости от величины Кн выделяют три зоны: критическую (Кн > 0,8); подкритическую (0,6 < Кн < 0,8); резервную (Кн < 0,6).



# Глава 2

Методы решения задач оперативного управления для автономной системы сетевой структуры

2.2 Целочисленное линейное программирование - метод ветвей и границ

Метод ветвей и границ — общий алгоритмический метод для нахождения оптимальных решений различных задач оптимизации, особенно дискретной и комбинаторной оптимизации. По существу, метод является комбинаторным (алгоритм перебора) с отсевом подмножеств множества допустимых решений, не содержащих оптимальных решений. Его суть заключается в упорядоченном переборе вариантов и рассмотрении лишь тех из них, которые оказываются по определенным признакам перспективными, и отбрасывании бесперспективных вариантов.

Метод ветвей и границ состоит в следующем: множество допустимых решений (планов) некоторым способом разбивается на подмножества, каждое из которых этим же способом снова разбивается на подмножества. Процесс продолжается до тех пор, пока не получено оптимальное целочисленное решение исходной задачи.

Метод был впервые предложен Ленд и Дойг в 1960 г. для решения задач целочисленного линейного программирования.

2.2.1 Общее описание

Общая идея метода может быть описана на примере поиска минимума и максимума функции f(x) на множестве допустимых значений x. Функция f и x могут быть произвольной природы. Для метода ветвей и границ необходимы две процедуры: ветвление и нахождение оценок (границ).

Процедура ветвления состоит в разбиении области допустимых решений на подобласти меньших размеров. Процедуру можно рекурсивно применять к подобластям. Полученные подобласти образуют дерево, называемое деревом поиска или деревом ветвей и границ. Узлами этого дерева являются построенные подобласти.

Процедура нахождения оценок заключается в поиске верхних и нижних границ для оптимального значения на подобласти допустимых решений.

В основе метода ветвей и границ лежит следующая идея (для задачи минимизации): если нижняя граница для подобласти A дерева поиска больше, чем верхняя граница какой-либо ранее просмотренной подобласти B, то A может быть исключена из дальнейшего рассмотрения (правило отсева). Обычно, минимальную из полученных верхних оценок записывают в глобальную переменную m; любой узел дерева поиска, нижняя граница которого больше значения m, может быть исключен из дальнейшего рассмотрения.

Если нижняя граница для узла дерева совпадает с верхней границей, то это значение является минимумом функции и достигается на соответствующей подобласти.

2.2.2 Применение

Метод используется для решения некоторых NP-трудных задач, такие как:

Задача коммивояжера

Задача о ранце

2.2.3 Алгоритм решения

Первоначально находим симплексным методом или методом искусственного базиса оптимальный план задачи без учета целочисленности переменных. Пусть им является план X0. Если среди компонент этого плана нет дробных чисел, то тем самым найдено искомое решение данной задачи и Fmax = F(Xo).

Если же среди компонент плана X0 имеются дробные числа, то X0 не удовлетворяет условию целочисленности и необходимо осуществить упорядоченный переход к новым планам, пока не будет найдено решение задачи. Покажем, как это можно сделать, предварительно отметив, что F(X0) ≥ F(X) для всякого последующего плана X.

Предполагая, что найденный оптимальный план X0 не удовлетворяет условию целочисленности переменных, тем самым считаем, что среди его компонент есть дробные числа. Пусть, например, переменная  приняла в плане X0 дробное значение. Тогда в оптимальном целочисленном плане ее значение будет по крайней мере либо меньше или равно ближайшему меньшему целому числу , либо больше или равно ближайшему большему целому числу +1. Определяя эти числа, находим симплексным методом решение двух задач линейного программирования:





Найдем решение задач линейного программирования (I) и (II). Очевидно, здесь возможен один из следующих четырех случаев:

1. Одна из задач неразрешима, а другая имеет целочисленный оптимальный план. Тогда этот план и значение целевой функции на нем и дают решение исходной задачи.

2. Одна из задач неразрешима, а другая имеет оптимальный план, среди компонент которого есть дробные числа. Тогда рассматриваем вторую задачу и в ее оптимальном плане выбираем одну из компонент, значение которой равно дробному числу, и строим две задачи, аналогичные задачам (I) и (II).

3. Обе задачи разрешимы. Одна из задач имеет оптимальный целочисленный план, а в оптимальном плане другой задачи есть дробные числа. Тогда вычисляем значения целевой функции на этих планах и сравниваем их между собой. Если на целочисленном оптимальном плане значение целевой функции больше или равно ее значению на плане, среди компонент которого есть дробные числа, то данный целочисленный план является оптимальным для исходной задачи и он вместе со значением целевой функции на нем дает искомое решение.

Если же значение целевой функции больше на плане, среди компонент которого есть дробные числа, то следует взять одно из таких чисел и для задачи, план которой рассматривается, необходимо построить две задачи, аналогичные (I) и (II).

4. Обе задачи разрешимы, и среди оптимальных планов обеих задач есть дробные числа. Тогда вычисляем значение целевой функции на данных оптимальных планах и рассматриваем ту из задач, для которой значение целевой функции является наибольшим. В оптимальном плане этой задачи выбираем одну из компонент, значение которой является дробным числом, и строим две задачи, аналогичные (I) и (II).

Таким образом, описанный выше итерационный процесс может быть представлен в виде некоторого дерева, на котором исходная вершина отвечает оптимальному плану Х0 задачи (1)-(3), а каждая соединенная с ней ветвью вершина отвечает оптимальным планам задач (I) и (II). Каждая из этих вершин имеет свои ветвления. При этом на каждом шаге выбирается та вершина, для которой значение функции является наибольшим. Если на некотором шаге будет получен план, имеющий целочисленные компоненты, и значение функции на нем окажется больше или равно, чем значение функции в других возможных для ветвления вершинах, то данный план является оптимальным планом исходной задачи целочисленного программирования и значение целевой функции на нем является максимальным.

Итак, процесс нахождения решения задачи целочисленного программирования (1)-(4) методом ветвей и границ включает следующие основные этапы:

1). Находят решение задачи линейного программирования (1)-(3).

2). Составляют дополнительные ограничения для одной из переменных, значение которой в оптимальном плане задачи (1)-(3) является дробным числом.

3). Находят решение задач (I) и (II), которые получаются из задачи (1)-(3) в результате присоединения дополнительных ограничений.

4). В случае необходимости составляют дополнительные ограничения для переменной, значение которой является дробным, формулируют задачи, аналогичные задачам (I) и (II), и находят их решение.

Итерационный процесс продолжают до тех пор, пока не будет найдена вершина, соответствующая целочисленному плану задачи (1)-(3) и такая, что значение функции в этой вершине больше или равно значению функции в других возможных для ветвления вершинах.

Описанный выше метод ветвей и границ имеет более простую логическую схему расчетов, чем метод Гомори.

В узлах метода ветвей и границ используется симплекс-метод.

Главный недостаток алгоритма метода ветвей и границ заключается в необходимости полностью решать задачи линейного программирования, ассоциированные с каждой из вершин многогранника допустимых решений. Для задач большой размерности это требует значительных и, в известной степени, неоправданных с практической точки зрения затрат времени.

3.1 Перераспределение средств

Оптимизация основана на перераспределении ресурсов из резервной зоны в критическую так, чтобы время выполнения всего комплекса стало минимальным. Переброска ресурсов возможна только между работами, у которых время их выполнения полностью или в большей своей части перекрывается. Снимая часть персонала и других ресурсов с резервной работы и направляя их на критическую работу, мы удлиняем продолжительность выполнения первой работы и сокращаем продолжительность второй.

При выполнении перераспределения ресурсов необходимо учитывать, что из-за ограниченности фронта работ численность исполнителей по отдельно взятой работе не должна возрастать или уменьшаться более чем в 1.5 ... 2 раза.

3.2 Привлечение дополнительных средств

Оптимизация основана на привлечении дополнительных средствна работы критического пути так, чтобы общий срок выполнения работ был равен директивному, а расход дополнительных средств минимален.

Ход оптимизации следующий. Выбирается работа критического пути, у которой коэффициент роста затрат минимален и производится сокращение ее продолжительности до большей из следующих велечин:

а) своего минимально-возможного значения;

б) того промежуточного значения, при котором в сетевом графике параллельно данной работе появляется еще одна ветвь критического пути.

В случае (б) дальнейшее сокращение продолжительности одной работы не ведет к сокращению продолжительности критического пути, так как прежняя ветвь критического пути, проходившая через эту работу, исчезает. Теперь придется сокращать одновременно продолжительности двух работ, лежащих на старой и новой ветвях, критического пути, если окажется, что сумма их коэффициентов роста затрат минимальна.

Можно принять за правило, что претендентами на сокращение продолжительностей являются:

а) одиночные работы, если параллельно им не появляются новые критические пути в ходе самого сокращения;

б) две и большее число работ одновременно, лежащие на параллельных ветвях критического путей, существующих до начала сокращения работ или появляющихся в ходе такого сокращения.

В этом случае претендентов на сокращениение продолжительности подбирают по минимуму коэффициентов роста затрат одиночных работ и сумм коэффициентов работ, лежащих на параллельных ветвях критических путей.

# Заключение

Методы сетевого планирования и управления обеспечивают руководителей и исполнителей на всех участках работы обоснованной информацией, которая необходима им для принятия решений по планированию, организации и управлению.

Сетевые модели могут быть широко использованы на всех отечественных при разработке как долгосрочных, так и текущих планов. Сетевое планирование позволяет не только определять потребность различных производственных ресурсов в будущем, но и координировать их рациональный расходов настоящем. С помощью сетевых графиков можно соединить в единую систему все материальные, трудовые, финансовые и многие другие ресурсы и средства производства и в идеальных (планируемых), и в реальных (существующих) экономических условиях.

В ходе выполнения дипломной работы были выполнены все поставленные цели: рассмотрены понятия сетевого планирования, по изученным правилам построен сетевой график, произведен расчет сетевого графика табличным методом. А так же выполнено условие оперативного управления.

# Список литературы

