

**РЕСПУБЛИКА УЗБЕКИСТАН**  
**НАВОИЙСКИЙ ГОРНО-МЕТАЛЛУРГИЧЕСКИЙ КОМБИНАТ**  
**НАВОИЙСКИЙ ГОСУДАРСТВЕННЫЙ ГОРНЫЙ ИНСТИТУТ**  
**“ЭНЕРГО - МЕХАНИЧЕСКИЙ” факультет**  
**кафедра “ЭЛЕКТРОЭНЕРГЕТИКА”**

**“Допущен к защите по выпускной работе”**

**“ \_\_\_\_\_ ” \_\_\_\_\_ 2014 г**

**Зав кафедрой \_\_\_\_\_ доц. Товбаев А.Н**

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

**Расчетно - пояснительная записка**

**Тема      ИССЛЕДОВАНИЕ РЕЖИМОВ РАБОТЫ СИСТЕМ  
ЭЛЕКТРОСНАБЖЕНИЯ**

**Выпускник** гр№ 2-10 ЭЭ, Бердикулов Ислон  
(группа, подпись    Ф.И.О)

**Руководитель**  
**выпускной работы** Ст.преп. Эшев Хамдам Хазратович  
(должность, звание, подпись, Ф.И.О)

**Консультанты:**  
**Экономическая часть** \_\_\_\_\_  
(должность, звание, подпись, Ф.И.О)

**Охрана труда** \_\_\_\_\_  
(должность, звание, подпись, Ф.И.О)

**Рецензент** \_\_\_\_\_  
(должность, звание, подпись, Ф.И.О)

**Норма контроля** \_\_\_\_\_  
(должность, звание, подпись, Ф.И.О)

**Зав. кафедрой** \_\_\_\_\_ **доц. Товбаев А.Н**

# **ДИПЛОМНАЯ РАБОТА**

по специальности “Системы управления производством  
и распределением электроэнергии”

На тему: **ИССЛЕДОВАНИЕ РЕЖИМОВ РАБОТЫ СИСТЕМ  
ЭЛЕКТРОСНАБЖЕНИЯ**

Студент

Руководитель проекта

Нормоконтроль

Рецензент

Проект рассмотрен на заседании кафедры и допущен к защите в ГЭК

Зав. кафедрой

**"УТВЕРЖДАЮ"**

Зав. кафедрой

\_\_\_\_\_  
(подпись, дата)**ЗАДАНИЕ  
на аттестационную работу**\_\_\_\_\_  
(фамилия, имя, отчество)**1. Тема работы: Исследование режимов работы систем электроснабжения**Исследование режимов работы электрических сетей ОАО "ММК" и разработка адаптивной системы управления режимами электропотребления**Тема утверждена приказом по****2. Цель исследования**Определение оптимальных режимов работы электрических сетей. Разработка программного обеспечения для расчета оптимальных режимов работы электрических сетей.**3. Исходные данные (характеристика объекта, условий исследования и др.)**Однолинейные схемы замещения электрических сетей. Графики нагрузок подстанций.**4. Основные задачи исследования:**Определить диапазон изменения параметров электрических сетей и нагрузок в рабочих и типовых аварийных режимах.Исследовать и выбрать наиболее эффективные методы оптимизации режимов сетей.**5. Срок представления работы к защите** \_\_\_\_\_**6. Дата выдачи задания** - \_\_\_\_\_**Научный руководитель** \_\_\_\_\_  
(подпись)**Задание принял к выполнению** \_\_\_\_\_  
(подпись)

## КАЛЕНДАРНЫЙ ПЛАН

(фамилия, имя, отчество студента)

| №<br>п | Название этапа работы                                                                                         | Срок выполнения<br>этапа | Примечание |
|--------|---------------------------------------------------------------------------------------------------------------|--------------------------|------------|
| 1.     | Получение задания на дипломную работу.                                                                        |                          |            |
| 2.     | Обзор технической литературы по теме работы.                                                                  |                          |            |
| 3.     | Изучение и описание системы электроснабжения ММК                                                              |                          |            |
| 4.     | Исследование режимов работы системы электроснабжения ММК                                                      |                          |            |
| 5.     | Разработка программного обеспечения для решения задачи оптимизации режимов электроснабжения                   |                          |            |
| 6.     | Составление полной схемы замещения системы электроснабжения ММК                                               |                          |            |
| 7.     | Расчет и оптимизация режимов электроснабжения ММК                                                             |                          |            |
| 8.     | Разработка структурной схемы автоматизированной системы оптимального управления режимами электроснабжения ММК |                          |            |
| 9.     | Выбор элементной базы для реализации автоматизированной системы оптимального управления                       |                          |            |
| 10.    | Написание пояснительной записки.<br>Создание слайдов для доклада                                              |                          |            |
| 11.    | Предварительная защита                                                                                        |                          |            |
| 12.    | Коррекция работы по результатам предзащиты                                                                    |                          |            |
| 13.    | Окончательное оформление пояснительной записки и слайдов.                                                     |                          |            |
| 14.    | Защита дипломной работы                                                                                       |                          |            |

Студент

\_\_\_\_\_

Руководитель проекта

\_\_\_\_\_

## РЕФЕРАТ

Пояснительная записка: 95 страниц, 25 рисунков, 21 таблица, 3 приложения, 14 источников.

Объект исследования: электрическая сеть ОАО ММК

Рассмотрены вопросы оптимизации текущего режима электропотребления по реактивной мощности и разработки адаптивной системы управления режимами электропотребления.

Разработано программное обеспечение по определению оптимальных режимов работы электрических сетей.

Предложена система автоматизации процесса распределения реактивной мощности, реализованная на оборудовании фирмы Allen-Bradley.

Разработанная система автоматизации производит сбор необходимых параметров, рассчитывает текущий режим, определяет оптимальные мощности компенсирующих устройств и передает управляющие воздействия на устройства, регулирующие мощности компенсирующих устройств.

**Оптимизация, Электропотребление**

**Реактивная мощность**

**Активная мощность**

**Потери мощности**

**Автоматизация**

# СОДЕРЖАНИЕ

|                                                                                                               |           |
|---------------------------------------------------------------------------------------------------------------|-----------|
| <b>ВВЕДЕНИЕ.....</b>                                                                                          | <b>8</b>  |
| <b>1 ИССЛЕДОВАНИЕ МЕТОДОВ ОПТИМИЗАЦИИ.....</b>                                                                | <b>14</b> |
| <b>1.1 ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ ОПТИМИЗАЦИИ.....</b>                                                    | <b>14</b> |
| <b>1.2 МАТЕМАТИЧЕСКАЯ МОДЕЛЬ.....</b>                                                                         | <b>14</b> |
| <b>1.3 МЕТОДЫ РЕШЕНИЯ ОПТИМИЗАЦИОННЫХ ЗАДАЧ.....</b>                                                          | <b>15</b> |
| <b>1.3.1 Общая характеристика методов решения задач<br/>                нелинейного программирования.....</b> | <b>16</b> |
| <b>1.4 МЕТОДЫ ПРЯМОГО ПОИСКА.....</b>                                                                         | <b>17</b> |
| <b>1.4.1 Метод Хука-Дживса.....</b>                                                                           | <b>18</b> |
| 1.4.1.1 Исследующий поиск.....                                                                                | 18        |
| 1.4.1.2 Поиск по образцу. ....                                                                                | 19        |
| 1.4.1.3 Описание алгоритма метода.....                                                                        | 19        |
| <b>1.4.2 Метод комплексов.....</b>                                                                            | <b>20</b> |
| <b>1.4.3 Методы случайного поиска.....</b>                                                                    | <b>21</b> |
| <b>1.4.4 Метод покоординатного спуска.....</b>                                                                | <b>23</b> |
| <b>1.5 ГРАДИЕНТНЫЕ МЕТОДЫ.....</b>                                                                            | <b>23</b> |
| <b>1.5.1 Градиентный метод с постоянным шагом.....</b>                                                        | <b>24</b> |
| <b>1.5.2 Метод скорейшего спуска.....</b>                                                                     | <b>25</b> |
| <b>1.5.3 Метод проектирования градиента.....</b>                                                              | <b>27</b> |
| <b>1.6 МЕТОД ШТРАФНЫХ ФУНКЦИЙ.....</b>                                                                        | <b>29</b> |
| <b>1.7 МЕТОДЫ ПОЛИНОМИАЛЬНОЙ АППРОКСИМАЦИИ.....</b>                                                           | <b>30</b> |
| <b>1.7.1 Квадратичная аппроксимация.....</b>                                                                  | <b>31</b> |
| 1.7.1.1 Метод Пауэлла .....                                                                                   | 32        |
| <b>1.7.2 Кубическая интерполяция.....</b>                                                                     | <b>33</b> |
| <b>1.7.3 Квадратичные функции.....</b>                                                                        | <b>35</b> |
| <b>1.8 МЕТОД НЕЛДЕРА-МИДА.....</b>                                                                            | <b>36</b> |
| <b>1.9 МЕТОД НЕОПРЕДЕЛЕННЫХ МНОЖИТЕЛЕЙ ЛАГРАНЖА.....</b>                                                      | <b>40</b> |
| <b>1.10 ВЫБОР МЕТОДА ОПТИМИЗАЦИИ.....</b>                                                                     | <b>42</b> |
| <b>2 РАЗРАБОТКА МЕТОДА ОПТИМИЗАЦИИ ПО РЕАКТИВНОЙ<br/>МОЩНОСТИ.....</b>                                        | <b>43</b> |
| <b>3 РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ МЕТОДА<br/>ОПТИМИЗАЦИИ .....</b>                                     | <b>49</b> |
| <b>4 РАЗРАБОТКА АДАПТИВНОЙ СИСТЕМЫ УПРАВЛЕНИЯ<br/>РЕЖИМАМИ ЭЛЕКТРОПОТРЕБЛЕНИЯ.....</b>                        | <b>55</b> |

|                                                                                      |    |
|--------------------------------------------------------------------------------------|----|
| <b>4.1 ФУНКЦИИ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ</b> .....                                  | 55 |
| <b>4.2 ОПИСАНИЕ РАБОТЫ СИСТЕМЫ</b> .....                                             | 55 |
| 4.2.1 Ввод системы в работу.....                                                     | 55 |
| 4.2.2 Работа системы в нормальном режиме.....                                        | 56 |
| 4.2.3 Работа системы в случае изменения конфигурации сети ...                        | 57 |
| <b>4.3 ТРЕБОВАНИЯ К ОБОРУДОВАНИЮ И ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ</b> .....                | 58 |
| <b>4.4 ВЫБОР ОБОРУДОВАНИЕ ДЛЯ АДАПТИВНОЙ СИСТЕМЫ</b> .....                           | 60 |
| 4.4.1 Учет электроэнергии.....                                                       | 60 |
| 4.4.1.1 Устройства мониторинга PowerMonitor .....                                    | 60 |
| 4.4.1.2 Программное обеспечение RSEnergy для контроля электропотребления.....        | 61 |
| 4.4.2 Процессор для диспетчерского пункта.....                                       | 62 |
| 4.4.3 Сервер связи.....                                                              | 63 |
| 4.4.3.1 Основные преимущества.....                                                   | 64 |
| 4.4.3.2 Минимальные требования RSLinx: .....                                         | 65 |
| 4.4.3.3 Различия между разными версиями программного обеспечения RSLinx .....        | 65 |
| 4.4.3.4 Версия программного обеспечения RSLinx Lite .....                            | 66 |
| 4.4.3.5 Версия программного обеспечения RSLinx OEM.....                              | 66 |
| 4.4.3.6 Версия программного обеспечения RSLinx Professional.....                     | 67 |
| 4.4.3.7 Версия программного обеспечения RSLinx Gateway .                             | 68 |
| 4.4.3.8 Версия программного обеспечения RSLinx SDK.....                              | 69 |
| 4.4.3.9 Графические функции SuperWho и RSWho .....                                   | 69 |
| <b>5 ИССЛЕДОВАНИЕ И ПОЛУЧЕНИЕ ОПТИМАЛЬНЫХ РЕЖИМОВ ДЛЯ ОАО "ММК ИМ. ИЛЬИЧА"</b> ..... | 71 |
| <b>5.1 РАСЧЕТ ПАРАМЕТРОВ СХЕМЫ ЗАМЕЩЕНИЯ</b> .....                                   | 71 |
| 5.1.1 Теоретические положения.....                                                   | 71 |
| 5.1.2 Расчет параметров схем замещения линий.....                                    | 73 |
| 5.1.3 Расчет параметров схем замещения трансформаторов.....                          | 73 |
| <b>5.2 РАСЧЕТ СЕТИ ПРИ РАЗЛИЧНЫХ НАГРУЗКАХ</b> .....                                 | 80 |
| <b>ВЫВОДЫ</b> .....                                                                  | 82 |
| <b>ПЕРЕЧЕНЬ ССЫЛОК</b> .....                                                         | 83 |
| <b>Приложение А</b> .....                                                            | 84 |
| <b>Приложение Б</b> .....                                                            | 93 |
| <b>Приложение В</b> .....                                                            | 96 |

## ВВЕДЕНИЕ

При исследовании режимов электрических сетей необходимо обратить особое внимание на явления, связанные с передачей реактивной мощности по сети, а также на способы ее компенсации.

В отличие от активной мощности реактивная мощность, потребляется элементами сети и электроприемниками в соизмеримых количествах. При этом она может генерироваться не только на электрических станциях, но и в сети. В частности, генерация реактивной мощности емкостью линий является вынужденной.

Реактивная мощность является практически удачной формой учета условий протекания периодических процессов в цепи переменного тока. Поскольку для обеспечения условий их протекания при допустимых параметра режима приходится применять специальные компенсирующие устройства, то возникает задача их наивыгоднейшего использования в условиях эксплуатации сети.

При решении этой задачи целесообразно прежде всего выяснить, с какими дополнительными явлениями связана передача реактивной мощности по элементам сети и какое влияние эти явления оказывают на технико-экономические показатели работы систем электроснабжения.

Как известно, передача реактивной мощности приводит к увеличению потерь напряжения в сети. С передачей реактивной мощности непосредственно связано увеличение нагрузки в соответствующих элементах сети.

Отсюда следует также и увеличение потерь активной мощности в элементах системы электроснабжения, которое должно учитываться в балансе по системе, т. е. компенсироваться соответствующей дополнительной установленной мощностью оборудования электрических станций.

Одновременно увеличиваются потери энергии за любой промежуток времени. Дополнительный расход электроэнергии означает дополнительный расход энергоносителей, практически топлива, что связано с дополнительными денежными и материальными расходами.

Это означает, что для выполнения поставленной вначале задачи в действительности требуется генерация соответственно большей реактивной мощности, т. е. практически установка дополнительных компенсирующих устройств.

Следует обратить внимание и на вторичное явление. Указанное выше увеличение потерь напряжения при неограниченной величине высшего допустимого напряжения приводит к снижению его низшего значения у приемного конца сети. А это приводит к увеличению значений токов при тех же значениях передаваемой мощности и к дополнительной потере активной и реактивной мощности, а также к дополнительной потере энергии.

При наличии соответствующих компенсирующих устройств целесообразно компенсировать реактивную мощность на месте, по возможности устраняя передачу ее по элементам сети на большие расстояния. Однако надо иметь в виду, что включение в работу новых устройств или увеличение нагрузки уже работающих иногда приводит к увеличению потерь активной мощности в них. Наивыгоднейшее решение заранее может быть неизвестно и получается путем расчета.

В распределительных сетях, особенно в промышленности, обычно имеет место потребление реактивной мощности в больших количествах. Основными потребителями реактивной мощности являются асинхронные двигатели, только намагничивающий ток которых составляет от 40 до 60% номинального.

Компенсирующие устройства (батареи конденсаторов), устанавливаемые в распределительных сетях, могут быть использованы для регулирования напряжения. Это сокращает потребность в местных регулирующих устройствах. Поэтому экономичность работы компенсирующих устройств в распределительных сетях следует рассматривать с учетом воздействия на режим напряжений.

При этом потери энергии в сети не обязательно получаются наименьшими, так же как и в случае распределения активной мощности между электрическими станциями в энергетической системе. Наибольшим должен быть результирующий экономический эффект.

Другое положение получается при использовании реактивной мощности, генерируемой синхронными двигателями, которые могут быть установлены в промышленной распределительной сети. Генерация реактивной мощности синхронными двигателями приводит к аналогичном эффекту регулирования напряжения, но связана со значительными потерями активной мощности, а следовательно, и энергии в самих двигателях. Поэтому регулирование напряжения автоматическим изменением тока возбуждения синхронных двигателей не всегда экономически обосновано. Обычно

использование для этого синхронных двигателей малой мощности экономически не оправдывается.

В питающей сети, на всех приемных подстанциях которой имеются регулирующие устройства с достаточным регулировочным диапазоном, распределение реактивной мощности можно осуществлять по условиям экономичности работы самой питающей сети. Определяющими здесь являются условия минимума потерь активной мощности в сети при заданных ограничениях по наибольшему допустимому напряжению и рабочей реактивной мощности источников питания.

В послеаварийных режимах перераспределением реактивной мощности в сети часто удается улучшить параметры режима. При этом экономичность режима приходится рассматривать как факт второстепенный.

Комплексная проблема определения оптимальных условий эксплуатации энергосистем или энергообъединений должна решаться на основе использования экономического критерия минимизации приведенных народнохозяйственных затрат на производство и распределение электрической и тепловой энергии. Решение этой проблемы в настоящее время осуществляется на основе рассмотрения ряда взаимосвязанных задач, условно представленных на рисунке ниже.

Каждая из перечисленных задач характеризуется своими частичным экономическим критерием и математической моделью поведения энергосистемы, которым отвечают определенные алгоритмы, наиболее полно учитывающие специфику задачи. В процессе решения используются различные исходные данные, в значительной мере вероятностные и неравноточные.

Под задачей оптимизации текущего режима энергосистемы или энергообъединения понимают наивыгоднейшее распределение генерируемых активных и реактивных мощностей между электростанциями, а также другими регулируемыми источниками реактивной мощности синхронными компенсаторами, управляемыми реакторами и батареями статических конденсаторов, которому отвечает минимум эксплуатационных издержек И на производство и распределение электрической и тепловой энергии в топливном или стоимостном выражениях:

$$И(Z)=\min$$

Целевая функция И зависит от вектора переменных Z, включающего в себя всю совокупность параметров режима.

## Взаимосвязь различных задач оптимизации режимов



В зависимости от принятой математической модели оптимизации, определяемой частичной задачей оптимизации и допущениями при составлении модели, в состав  $Z$  могут входить активные и реактивные

мощности электростанций  $P_S + jQ_S$ , коэффициенты трансформации (в общем случае комплексные)  $k'_t + jk''_t$ , модули напряжений  $U_t$  в некоторых или во всех узлах расчетной схемы, а при необходимости и другие параметры режима и оборудования.

Составляющие вектора  $Z$  связаны между собой рядом конкретных условий, накладываемых свойствами электрической сети, техническими характеристиками и условиями надежной работы оборудования. Сюда относятся обеспечение баланса мощностей, ограничение напряжений в узлах, ограничение тока и передаваемой мощности по линиям электропередачи, допустимость режимов по устойчивости параллельной работы электростанций, ограничение мощности электростанций по характеристикам оборудования и т. п.

В такой общей постановке задача оптимизации текущего режима является многоэкстремальной задачей нелинейного математического программирования и для произвольного вида функции  $I(Z)$  строгие методы ее решения не разработаны.

Из изложенного следует, что определения оптимального рабочего режима электрической сети в процессе ее текущей эксплуатации нужно значительное количество информации о параметрах режима и требуется выполнение достаточно сложных расчетов по ее обработке и получению ответа. В некоторых случаях задача должна решаться одновременно для всей энергетической системы. Для этого требуется достаточно сложное программное и аппаратное обеспечение, осуществляющее получение и обработку информации, а также управление всеми автоматизированными устройствами, имеющимися в системе.

В своем дипломном проекте я оптимизирую текущий режим энергосистемы, а именно, минимизирую целевую функцию путем решения задачи нелинейного программирования на языке программирования C++.

Задача безусловной минимизации (минимизации без ограничений) состоит в поиске минимума  $\min f(x)$ , где функция  $f: R^n \rightarrow R$  - является по крайней мере непрерывной. Процедуры безусловной минимизации подразделяются на 3 категории:

- оперирующие функцией одной переменной;
- работающие с функцией нескольких переменных;
- использующие нелинейный метод наименьших квадратов.

В случае функции одной переменной предполагается, что на исследуемом отрезке она имеет один экстремум. В противном случае осуществляется поиск локального минимума.

Поиск минимума функции нескольких переменных можно выполнить квазиньютоновским методом, модифицированным алгоритмом Ньютона, методом сопряженных градиентов и методом деформируемого многогранника.

Перечисленные процедуры обеспечивают поиск локального минимума. Если же функция имеет несколько локальных минимумов и необходимо найти наилучший, то следует испытать разные начальные точки и интервалы поиска. С процедурами, использующими только значения функции, следует употреблять двойную точность. Также полезно использовать процедуры контроля производной, обеспечивающие проверку работы пользовательских процедур, оценивающих производные.

Как уже указывалось, в настоящее время не существует единообразного подхода к задаче оптимизации мгновенного режима энергосистемы. Все многообразие практических методов использования ЦВМ можно классифицировать по некоторым главным направлениям. Основными задачами расчетов могут являться:

а) комплексная оптимизация распределения активных и реактивных генерируемых мощностей и коэффициентов трансформации по условию минимума суммарного расхода или стоимости топлива в системе;

б) оптимальное распределение активных мощностей между электростанциями с приближенным учетом потерь в сети по условию минимума суммарного расхода или стоимости топлива;

в) оптимальное распределение реактивных мощностей между электростанциями и синхронными компенсаторами и выбор оптимальных коэффициентов трансформации регулируемых трансформаторов по минимуму потерь в сети.

В расчетах в качестве математической модели установившегося режима используются либо полные уравнения установившегося режима, обычно уравнения узловых напряжений в форме баланса мощностей в узлах, либо упрощенные уравнения баланса активной мощности в целом по системе

$$\sum_s P_s - \sum_i P_i - \pi = 0.$$

## **1 Исследование методов оптимизации**

### **1.1 Основные понятия и определения оптимизации**

Показатель, по величине которого оценивают, является ли решение оптимальным, называется критерием оптимальности.[1] В качестве критерия оптимальности наиболее часто принимается экономический критерий, представляющий собой минимум затрат (финансовых, сырьевых, энергетических, трудовых) на реализацию поставленной задачи. При заданной или ограниченной величине указанных затрат экономический критерий выражается в получении максимальной прибыли.

В электроэнергетике в зависимости от требований поставленной задачи могут применяться и другие критерии оптимальности, в частности:

- критерий надежности электроснабжения;
- критерий качества электроэнергии;
- критерий наименьшего отрицательного воздействия на окружающую среду (экологический критерий).

Решение оптимизационной задачи включает в себя следующие этапы:

- сбор исходной информации (исходных данных);
- составление математической модели, под которой понимается формализованное математическое описание решаемой задачи;
- выбор метода решения, определяемого видом математической модели;
- выполнение математических вычислений, поручаемое, как правило, компьютеру;
- анализ решения задачи.

### **1.2 Математическая модель**

Математическая модель – формализованное математическое описание оптимизационной задачи.[1,2] Математическая модель включает в себя:

- целевую функцию;
- ограничения;
- граничные условия.

Целевая функция представляет собой математическую запись критерия оптимальности. При решении оптимизационной задачи ищется экстремум целевой функции, например минимальные затраты или максимальная прибыль. Обобщенная запись целевой функции имеет следующий вид:

$$Z \ x_1, x_2, \dots, x_n \rightarrow \text{extr} , \quad (1.1)$$



### 1.3.1 Общая характеристика методов решения задач нелинейного программирования

Когда целевая функция (1.1) и ограничения (1.2) нелинейны и для поиска точки экстремума нельзя или очень сложно использовать аналитические методы решения, тогда для решения задач оптимизации применяются методы нелинейного программирования. Как правило, при решении задач методами нелинейного программирования используются численные методы с применением ЭВМ [3,4,5,6].

В основном методы нелинейного программирования могут быть охарактеризованы как многошаговые методы или методы последующего улучшения исходного решения. В этих задачах обычно заранее нельзя сказать, какое число шагов гарантирует нахождение оптимального значения с заданной степенью точности. Кроме того, в задачах нелинейного программирования выбор величины шага представляет серьезную проблему, от успешного решения которой во многом зависит эффективность применения того или иного метода. Разнообразие методов решения задач нелинейного программирования как раз и объясняется стремлением найти оптимальное решение за наименьшее число шагов.

Большинство методов нелинейного программирования используют идею движения в  $n$ -мерном пространстве в направлении оптимума. При этом из некоторого исходного или промежуточного состояния  $U_k$  осуществляется переход в следующее состояние  $U_{k+1}$  изменением вектора  $U_k$  на величину  $DU_k$ , называемую шагом, т.е.

$$U_{k+1} = U_k + DU_k. \quad (1.4)$$

В ряде методов шаг, т.е. его величина и направление определяется как некоторая функция состояния  $U^k$

$$DU_k = f(U_k). \quad (1.5)$$

Следовательно, согласно (1.4) новое состояние  $U_k$ , получаемое в результате выполнения шага (1.5) может рассматриваться как функция исходного состояния  $U_k$

$$U_{k+1} = U_k + f(U_k). \quad (1.6)$$

В некоторых методах  $DU_k$  обусловлен не только состоянием  $U_k$ , но и рядом предшествующих состояний

$$DU_k = f(U_k, U_{k-1}, \dots, U_{k-2}), \quad (1.7)$$

$$U_{k+1} = U_k + f(U_k, U_{k-1}, \dots, U_{k-2}). \quad (1.8)$$

Естественно, что алгоритмы поиска типа (1.8) являются более общими и принципиально могут обеспечить более высокую сходимость к оптимуму,

т.к. используют большой объем информации о характере поведения оптимальной функции.

В настоящее время для решения подобных задач разработано значительное число методов, однако нельзя отдать предпочтение какому-либо одному. Выбор метода определяется сложностью объекта и решаемой задачей оптимизации.

Методы решения задач нелинейного программирования (условной многопараметрической оптимизации) подразделяют следующим образом:

- методы прямого поиска;
- градиентные методы;
- методы штрафных функций;
- методы полиномиальной аппроксимации.

#### **1.4 Методы прямого поиска**

Одними из методов нахождения минимума функции  $n$ -переменных являются методы прямого поиска. Методы прямого поиска являются методами, в которых используются только значения функции [1,7].

В методах прямого поиска ограничения учитываются в явном виде. Необходимость разработки этих методов связана с тем, что в инженерных приложениях часто приходится сталкиваться с случаями, когда целевые функции не заданы в явном виде. Эти методы строятся на интуитивных соображениях, не подкреплены строгой теорией и, следовательно, не гарантируется их сходимость. Несмотря на это, в силу своей логической простоты эти методы легко реализуются.

Перед непосредственным применением методов прямого поиска необходимо провести ряд мероприятий по подготовке задачи к решению, а именно

- исключить ограничения в виде равенств;
- определить начальную допустимую точку.

Простейший способ исключения ограничений в виде равенств заключается в решении его относительно одной из переменных с последующим исключением этой переменной путем подстановки полученного выражения в соотношения, описывающие задачу. При этом следует учитывать, что границы значений исключаемых переменных сохраняются в задаче в виде ограничений - неравенств.

Несмотря на то, что подстановка является самым простым способом исключения ограничений - равенств, не всегда оказывается возможным ее

осуществить. В этом случае проблема решается путем численного решения уравнения относительно зависимых переменных при заданных значениях независимых оптимизирующих переменных.

Для определения начальной допустимой точки целесообразно использовать процедуру случайного поиска, основная идея которого будет рассмотрена ниже.

После проведения процедуры подготовки задачи к решению следует отметить один из методов условной оптимизации [5,6]. Рассмотрим методы прямого поиска:

- модифицированный метод Хука-Дживса;
- метод комплексов;
- метод случайного поиска;
- метод покоординатного спуска.

#### **1.4.1 Метод Хука-Дживса**

Одним из методов прямого поиска есть метод Хука-Дживса[5,7], который был разработан в 1961г, но до сих пор является весьма эффективным и оригинальным. Метод Хука-Дживса характеризуется несложной стратегией поиска, относительной простотой вычислений и невысоким уровнем требований к памяти ЭВМ. Это один из первых алгоритмов, в котором при определении нового направления поиска учитывается информация, полученная на предыдущих итерациях. Процедура Хука-Дживса представляет собой комбинацию исследующего поиска с циклическим изменением переменных и ускоряющего поиска по образцу.

##### **1.4.1.1 Исследующий поиск**

Для проведения исследующего поиска необходимо знать величину шага, которая может быть различной для разных координатных направлений и изменяться в процессе поиска. Исследующий поиск начинается в некоторой исходной точке. Если значение целевой функции в пробной точке не превышает значение в исходной точке, то шаг поиска рассматривается как успешный. В противном случае надо вернуться в предыдущую точку и сделать шаг в противоположном направлении с последующей проверкой значения целевой функции. После перебора всех  $n$ - координат исследующий поиск завершается. Полученную в результате точку называют базовой точкой.

### 1.4.1.2 Поиск по образцу

Поиск по образцу [5,6] заключается в реализации единственного шага из полученной базовой точки вдоль прямой, соединяющей эту точку с предыдущей базовой точкой. Новая точка образца определяется в соответствии с формулой

$$X_p^{(k+1)} = X^{(k)} + (X^{(k)} - X^{(k-1)}). \quad (1.9)$$

Как только движение по образцу не приводит к уменьшению целевой функции, точка  $X_p^{(k+1)}$  фиксируется в качестве временной базовой точки и вновь проводится исследующий поиск. Если в результате получается точка с меньшим значением целевой функции, чем в точке  $X^{(k)}$ , то она рассматривается как новая базовая точка  $X^{(k+1)}$ . С другой стороны, если исследующий поиск неудачен, необходимо вернуться в точку  $X^{(k)}$  и провести исследующий поиск с целью выявления нового направления минимизации. В конечном счете возникает ситуация, когда такой поиск не приводит к успеху. В этом случае требуется уменьшить величину шага путем введения некоторого множителя и возобновить исследующий поиск. Поиск завершается, когда величина шага становится достаточно малой. Последовательность точек, получаемую в процессе реализации метода, можно записать в следующем виде:

- $X^{(k)}$  - текущая базовая точка,
- $X^{(k-1)}$  - предыдущая базовая точка,
- $X_p^{(k+1)}$  - точка, построенная при движении по образцу,
- $X^{(k+1)}$  - следующая (новая) базовая точка.

Приведенная последовательность характеризует логическую структуру поиска по методу Хука-Дживса.

### 1.4.1.3 Описание алгоритма метода

Шаг 1. Выбрать начальную базисную точку  $b_1$  и шаг длиной  $h_j$  для каждой переменной  $x_j$ ,  $j=1,2,\dots,n$ .

Шаг 2. Вычислить  $f(x)$  в базисной точке  $b_1$  с целью получения сведений о локальном поведении функции  $f(x)$ . Функция  $f(x)$  в базисной точке  $b_1$  находится следующим образом:

Вычисляется значение функции  $f(b)$  в базисной точке  $b_1$ .

Каждая переменная по очереди изменяется прибавлением длины шага. Таким образом, мы вычисляем значение функции  $f(b_1 + h_1 * e_1)$ , где  $e_1$  - единичный вектор в направлении оси  $x_1$ .

Если это приводит к уменьшению значения функции, то  $d_1$  заменяется на  $b_1 + h_1 * e_1$ . В противном случае вычисляется значение функции  $f(b_1 - h_1 * e_1)$ , и если ее значение уменьшилось, то  $b_1$  заменяем на  $b_1 - h_1 * e_1$ .

Если ни один из проделанных шагов не приводит к уменьшению значения функции, то точка  $b_1$  остается неизменной и рассматриваются изменения в направлении оси  $x_2$ , т.е. находится значение функции  $f(b_1 + h_2 * e_2)$  и т.д. Когда будут рассмотрены все  $n$ -переменных, мы будем иметь новую базисную точку  $b_2$ .

Если  $b_2 = b_1$ , т.е. уменьшение функции не было достигнуто, то исследование повторяется вокруг той же базисной точки  $b_1$ , но с уменьшенной длиной шага.

Если  $b_2 \neq b_1$ , то производится поиск по образцу.

Шаг 3. При поиске по образцу используется информация, полученная в процессе исследования, и минимизация функции завершается поиском в направлении, заданном образцом. Эта процедура производится следующим образом:

Разумно двигаться из базисной точки  $b_2$  в направлении  $b_2 - b_1$ , поскольку поиск в этом направлении уже привел к уменьшению значения функции. Поэтому вычислим функцию в точке образца

$$P_1 = b_1 + 2 * (b_2 - b_1). \quad (1.10)$$

В общем случае

$$P_i = b_i + 2 * (b_{(i+1)} - b_i). \quad (1.11)$$

Затем исследование следует продолжать вокруг точки  $P_1(P_i)$ .

Если наименьшее значение на шаге В,2 меньше значения в базисной точке  $b_2$  (в общем случае  $b_{(i+1)}$ ), то получают новую базисную точку  $b_3$  ( $b_{(i+2)}$ ), после чего следует повторить шаг В,1. В противном случае не производить поиск по образцу из точки  $b_2(b_{(i+1)})$ , а продолжить исследования в точке  $b_2(b_{(i+1)})$ .

Шаг 4. Завершить этот процесс, когда длина шага (длины шагов) будет уменьшена до заданного малого значения.

#### 1.4.2 Метод комплексов

Алгоритм [7]:

Заданы границы значений всех переменных  $x_i^L$ ,  $x_i^U$ ,  $i=1,2,..., N$  (размерность задачи), допустимая точка  $x_0$ , параметр отображения  $a$  (рекомендуется  $a=1,3$ ) и параметры окончания вычислений  $\varepsilon$  и  $d$ .

Шаг 1. Построение начального комплекса, состоящего из  $P$  допустимых точек (рекомендуется  $P=2N$ ). Для каждой точки  $p = 1, 2, ..., P-1$

- случайным образом определить координаты  $x^p$ ;
- если  $x^p$  - недопустимая точка, то найти центр тяжести  $x_{\text{цт}}$  уже найденных точек и положить  $x^p = x^p + (x_{\text{цт}} - x^p)$ ; повторять процедуру до тех пор, пока  $x^p$  не станет допустимой;
- если  $x^p$  - допустимая точка, повторять до тех пор, пока  $p=P$ ;
- вычислить  $W(x^p)$  для  $p=0,1,\dots,P-1$ .

Шаг 2. Отражение комплекса:

- выбрать точку  $x^R$ , для которой  $W(x^R) = \max W(x^p) = W_{\max}$  (решается задача минимизации);
- найти центр тяжести  $x_{\text{цт}}$  и новую точку  $x^m = x_{\text{цт}} + a(x_{\text{цт}} - x^R)$ ;
- если  $x^m$  - допустимая точка и  $W(x^m) < W_{\max}$ , уменьшить в два раза расстояние между  $x^m$  и центром тяжести  $x_{\text{цт}}$ , продолжать поиск, пока  $W(x^m) < W_{\max}$ ;
- если  $x^m$  - допустимая точка и  $W(x^m) < W_{\max}$ , то перейти к шагу 4;
- если  $x^m$  - недопустимая точка, то перейти к шагу 3.

Шаг 3. Корректировка для обеспечения допустимости:

- если  $x_i^m < x_i^L$  (нижняя граница допустимой области), то положить  $x_i^m = x_i^L$ ;
- если  $x_i^m > x_i^U$  (верхняя граница допустимой области), то положить  $x_i^m = x_i^U$ ;
- если  $x^m$  - недопустимая точка, то уменьшить в два раза расстояние до центра тяжести; продолжать до тех пор, пока  $x^m$  не станет допустимой точкой.

Шаг 4. Проверка условий окончания вычислений:

- положить  $\bar{W} = \frac{1}{P} \sum_p W(x^p)$  и  $\bar{x} = \frac{1}{P} \sum_p x^p$ ;
- если  $\sum_p W(x^p) - \bar{W}^2 \leq \varepsilon$  и  $\sum_p x^p - \bar{x}^2 \leq \delta$ , то вычисления

прекратить; в противном случае перейти к шагу 2а.

### 1.4.3 Методы случайного поиска

Наиболее простой процедурой случайного поиска [3,5] является прямая выборочная процедура, заключающаяся в разыгрывании на ЭВМ последовательности точек с координатами

$$x_i = x_i^L + r_i (x_i^U - x_i^L), \quad i=1,2,\dots,N, \quad (1.12)$$

где  $r_i$  - случайная величина, равномерно распределенная на интервале  $[0,1]$ .

После проверки каждой точки на допустимость вычисляются значения целевой функции, которые сравниваются с наилучшим значением, полученным к данному моменту. Если текущая точка дает лучшее значение, то она запоминается, в противном - отбрасывается. Процесс прекращается после заданного числа итераций  $N$  или по исчерпанию заданного машинного времени. Для получения 90% доверительного интервала величиной  $\varepsilon_i (x_i^U - x_i^L)$ , где  $0 < \varepsilon < 1$ , для переменной  $x_i$  совместный случайный поиск требует  $2,3 \prod_{i=1}^N \varepsilon_i^{-1}$  испытаний. При  $N=5$ ,  $\varepsilon_i=0,01$  число испытаний оценивается в 2,3 1010, что исключает возможность непосредственного использования метода.

Значительного увеличения эффективности процедуры случайного поиска можно достигнуть путем группировки выборок в серии. При этом наилучшая точка в каждой серии используется как начальная точка следующей серии, точки которой уже выбираются из интервала меньшей величины. Данная процедура получила название выборки со сжатием пространства поиска. Рассмотрим более подробно ее алгоритм.

Заданы границы значений всех переменных  $x_i^L, x_i^U, i=1,2,..., N$  (размерность задачи), начальная допустимая точка  $x^0$  и интервал поиска  $D x^0 = x_i^U - x_i^L$ , количество серий  $Q$ , количество точек в серии  $P$  и параметр окончания вычислений  $\varepsilon$ . Для каждой из серий, начиная с  $q = 1$ , необходимо выполнить следующие действия.

Шаг 1. Для  $i = 1,2,...,N$  вычислить

$$x_i^p = x_i^{q-1} + r_i D x^{q-1}, \quad (1.13)$$

где  $r_i$  - случайная величина, равномерно распределенная на интервале  $[-0,5;0,5]$ .

Шаг 2.

- если  $x^p$  - недопустимая точка и  $p < P$ , то повторить шаг 1;
- если  $x^p$  - допустимая точка, то запомнить  $x^p$  и  $W(x^p)$  и
  - если  $p < P$ , то повторить шаг 1;
  - если  $p = P$ , то найти среди всех допустимых точек  $x^p$  точку с наименьшим значением  $W(x^p)$  и положить ее равной  $x^q$ .

Шаг 3. Уменьшить интервал поиска, полагая  $D \cdot x_i^q = \varepsilon_i \cdot D \cdot x_i^{q-1}$ .

Шаг 4.

Если  $q > Q$ , то закончить вычисления.

В противном случае увеличить  $q=q+1$  и продолжить вычисления, начиная с шага 1.

### 1.4.4 Метод покоординатного спуска

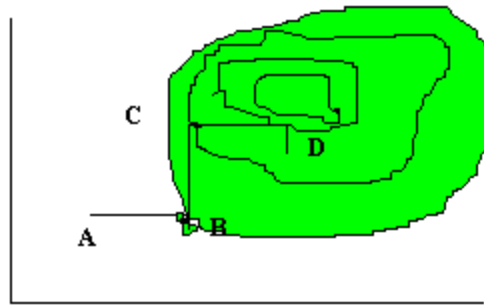


Рисунок 1.1 - Метод покоординатного спуска

Рассмотрим функцию двух переменных. Ее линии уровня представлены на рис. 1.1, а минимум лежит в точке  $(x_1^*, x_2^*)$ . Простейшим методом поиска является метод покоординатного спуска[3,4]. Из точки А произведем поиск минимума вдоль направления оси  $x_1$  и, таким образом, находим точку В, в которой касательная к линии постоянного уровня параллельна оси  $x_1$ . Затем, производя поиск из точки В в направлении оси  $x_2$ , получаем точку С, производя поиск параллельно оси  $x_2$ , получаем точку D, и т.д. Таким образом, мы приходим к оптимальной точке. Очевидным образом эту идею можно применить для функции  $n$  переменных.

Теоретически данный метод эффективен в случае единственного минимума функции. Но на практике он оказывается слишком медленным. Поэтому были разработаны более сложные методы, использующие больше информации на основании уже полученных значений функции.

### 1.5 Градиентные методы

Как следует из названия, эти методы решения нелинейных оптимизационных задач используют понятие градиента функции[3,5,7]. Градиентом функции  $Z(x_1, x_2, \dots, x_n)$  называется вектор

$$\text{grad}Z = \frac{\partial Z}{\partial x_1} \bar{i} + \frac{\partial Z}{\partial x_2} \bar{j} + \dots + \frac{\partial Z}{\partial x_n} \bar{k}, \quad (1.14)$$

где  $\bar{i}, \bar{j}, \dots, \bar{k}$  - единичные вектора (орты).

Величина этого вектора определяется по выражению

$$|\text{grad}Z| = \sqrt{\left(\frac{\partial Z}{\partial x_1}\right)^2 + \left(\frac{\partial Z}{\partial x_2}\right)^2 + \dots + \left(\frac{\partial Z}{\partial x_n}\right)^2}. \quad (1.15)$$

Из (1.14) и (1.15) видно, что функция, градиент которой определяется, должна быть дифференцируемой по всем  $n$  переменным.

Физический смысл градиента функции в том, что он показывает направление (1.14) и скорость (1.15) наибольшего изменения функции в рассматриваемой точке. Если в некоторой точке  $|\text{grad}Z|=0$ , функция в этой точке не изменяется (не возрастает и не убывает). Эта точка соответствует экстремуму функции.

### 1.5.1 Градиентный метод с постоянным шагом

Сущность градиентных методов решения нелинейных оптимизационных задач [1,5,7] поясним для случая отыскания абсолютного минимума функции двух переменных  $Z(x_1, x_2)$ , иллюстрируемого рис. 1.2. Этот минимум находится в точке с координатами  $x_{10}$  и  $x_{20}$ .

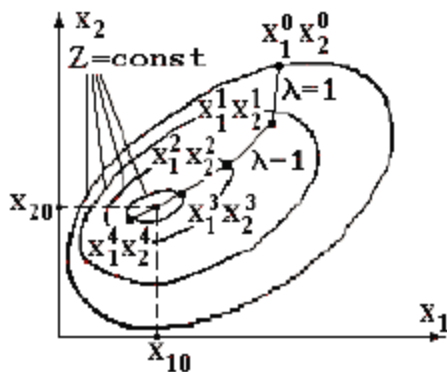


Рисунок 1.2 – Иллюстрация градиентного метода с постоянным шагом  $\lambda=1$

В соответствии с граничными условиями (1.3), в большинстве практических оптимизационных задач они принимают только положительные или нулевые значения, областью  $\Omega$  допустимых значений переменных будет первый квадрант системы координат  $x_1$  и  $x_2$ . в этой области произвольно выберем исходное (нулевое) приближение – точку с координатами  $x_1^0, x_2^0$ . значение целевой функции в этой точке составляет  $Z^0$ . В соответствии с выражением (1.15) вычислим в этой точке величину градиента функции  $Z$ .

Выполним шаг единичной длины ( $\lambda=1$ ) в направлении убывания функции  $Z$ . В результате выполненного шага получим первое приближение – точки с координатами  $x_1^1, x_2^1$ . Значение целевой функции в этой точке составляет  $Z^1$ .

Далее вычислительная процедура повторяется: последовательно получаем 2-е, 3-е и 4-е приближения – точки с координатами  $x_1^2, x_2^2; x_1^3, x_2^3$  и

$x_1^4, x_2^4$ . Значения целевой функции в этих точках соответственно составляют  $Z^2, Z^3$  и  $Z^4$ .

Из рис. 1.2 видно, что в результате вычислительного процесса последовательно осуществляется "спуск" к минимуму функции  $Z$ . Вычислительная процедура заканчивается, когда относительное изменение целевой функции на предыдущем  $i$ -м и последующем  $(i+1)$ -м шагах оказывается меньше заданной точности вычислений  $\varepsilon$ :

$$\frac{Z_i - Z_{i+1}}{Z_i} \leq \varepsilon. \quad (1.16)$$

Рассмотренная вычислительная процедура носит название градиентного метода с постоянным шагом. В этом методе все шаги выполнялись одинаковой длины  $\lambda=1$ . Метод достаточно прост. Основной его недостаток – большая вероятность заикливания вычислительного процесса в окрестности минимума функции  $Z$ . В соответствии с рис. 1.2 вычислительный процесс заиклится между точками с координатами  $x_1^3, x_2^3$  и  $x_1^4, x_2^4$ . При этом в качестве искомого решения следует принять одну из этих точек.

Для получения более точного результата необходимо выбрать шаг меньшей длины. При этом объем вычислений (количество шагов) увеличится.

Таким образом, точность и объем вычислений в градиентном методе с постоянным шагом определяются величиной этого шага.

### 1.5.2 Метод скорейшего спуска

Как было отмечено выше, при увеличении длины шага объем вычислений (количество шагов) уменьшается, однако уменьшается и точность определения минимума целевой функции. При уменьшении длины шага точность увеличивается, однако объем вычислений (количество шагов) возрастает.

Поэтому вопрос о выборе рациональной длины шага в градиентных методах является своего рода оптимизационной задачей. Один из способов определения оптимальной длины шага  $\lambda_{opt}$  иллюстрируется на рис. 1.3 и носит название метода скорейшего спуска [1,7].

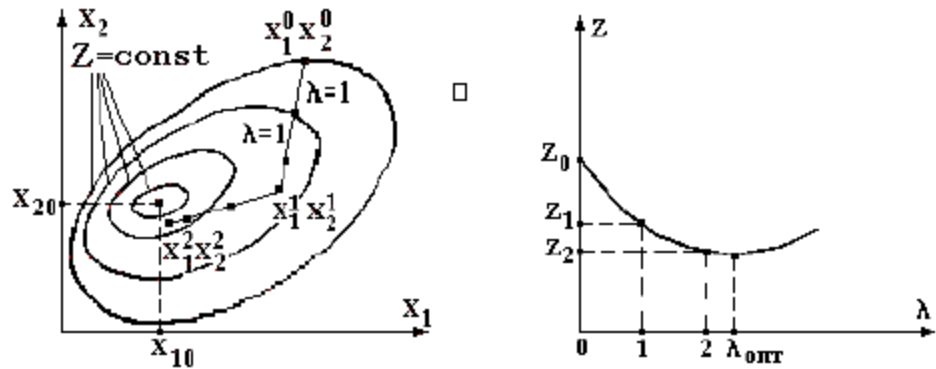


Рисунок 1.3 – Иллюстрация метода скорейшего спуска (а) и параболическая аппроксимация целевой функции для выбора оптимального шага (б)

В методе наискорейшего спуска желательно использовать рассмотренное свойство направления градиента. Поэтому, если мы находимся в точке  $x_i$  на некотором шаге процесса оптимизации, то поиск минимума функции осуществляется вдоль направления  $-\nabla f \ x_j$ . Данный метод является итерационным. На шаге  $i$  точка минимума аппроксимируется точкой  $x_i$ . Следующей аппроксимацией является точка

$$x_{i+1} = x_i - \lambda_i \nabla f \ x_i, \quad (1.17)$$

где  $\lambda_i$  - значение  $\lambda$ , минимизирующее функцию.

$$\varphi \ \lambda_i = f \left[ x_i - \lambda \nabla f \ x_i \right]. \quad (1.18)$$

Значение  $\lambda_i$  может быть найдено с помощью одного из методов одномерного поиска (например, методом квадратичной интерполяции).

В приложении приведена программа, позволяющая реализовать метод наискорейшего спуска. В ней множитель Лагранжа обозначен через  $h$ . Вектор  $d_i$  является единичным.

Для поиска минимума функции

$$\varphi \ \lambda = f \ x_i - \lambda d_i \quad (1.19)$$

в направлении  $d_i$  из точки  $x_i$  используется метод квадратичной интерполяции.

В точке  $x_i, \lambda=0$ , и мы выбираем длину шага  $\lambda$  такой, чтобы шаг "перекрыл" минимум функции  $\varphi(\lambda)$ . Производная

$$\frac{d\varphi}{d\lambda} = g(x_i - \lambda d_i)^T d_i. \quad (1.20)$$

Данный оператор `for(i=0;i<n;i++) g2+=g[i]*d[i];` - вычисляет выражение

$$\frac{d\varphi}{dh} = \nabla f^T x_0 - h d_i^T d = g^T x_0 - h d_i^T d. \quad (1.21)$$

Оператор `if (ff[2]>=ff[0] || g2>=0)` проверяет условие "перекрытия" минимума, которое выполняется при выполнении либо одного, либо другого условия. Если минимум не попал в отрезок  $(0, \lambda)$ , то  $\lambda$  удваивается, и это повторяется столько раз, сколько необходимо для выполнения условия "перекрытия".

Удостоверившись, что отрезок  $(0, \lambda)$  содержит минимум, в качестве третьей точки возьмем точку  $\lambda/2$ . Минимальную точку сглаживающего квадратичного полинома находим в соответствии с соотношением

$$\delta = \frac{1}{2} \frac{\gamma^2 - \alpha^2 f_\beta + \beta^2 - \gamma^2 f_\alpha + \alpha^2 - \beta^2 f_\gamma}{\gamma - \alpha f_\beta + \beta - \gamma f_\alpha + \alpha - \beta f_\gamma}, \quad (1.22)$$

что отражено следующими операторами

```
l[3]=h*(ff[1]-.75*ff[0]-.25*ff[2]);
```

```
l[3]/=2*ff[1]-ff[0]-ff[2];
```

```
Оператор for(i=0;i<n;i++)
```

```
{ x[i]=y[i]+l[0]*d[i]; y[i]=x[i]; }
```

производит присваивание  $x_{i+1}=x_i$ , и если  $|g(x_{i+1})|$  достаточно мало, то процесс заканчивается. В процессе поиска предполагается сходимость к экстремуму, поэтому для эффективности процедуры разумно уменьшить длину шага. При этом деление шага пополам выбрано произвольно.

В методе скорейшего спуска, по сравнению с градиентным методом с постоянным шагом, количество шагов меньше, точность получаемого результата выше, отсутствует закливание вычислительного процесса, однако объем вычислений на одном шаге больше.

### 1.5.3 Метод проектирования градиента

Рассмотренные выше градиентные методы предполагали отыскание абсолютного минимума целевой функции  $Z$ . При наличии в математической модели нелинейных ограничений ищется уже не абсолютный, а относительный минимум целевой функции  $Z$  [1].

Рассмотрим один из методов отыскания относительного минимума целевой функции, получивший название метода проектирования градиента.

Для упрощения алгоритма допустим, что имеется одно ограничение в виде линейного неравенства

$$ax_1 + bx_2 + c \geq 0. \quad (1.23)$$

При наличии указанного ограничения минимум целевой функции следует искать в области  $\Omega$ , расположенной по одну сторону от прямой  $ax_1 + bx_2 + c = 0$ , например выше этой прямой (рис. 1.4).

Начало вычислительной процедуры такое же, как и в предыдущих методах:

- в области  $\Omega$  принимается исходное (нулевое) приближение  $x_1^0, x_2^0$ ;
- вычисляется значение целевой функции в этой точке  $Z^0$ ;
- в соответствии с выражением (1.15) в этой точке вычисляется градиент целевой функции  $\text{grad } Z$ ;
- из исходной точки в направлении убывания целевой функции выполняется шаг.

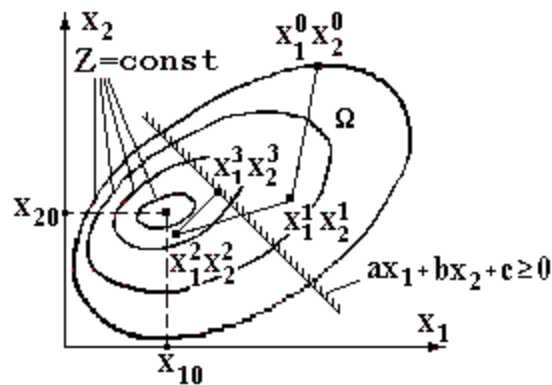


Рисунок 1.4 – Иллюстрация метода проектирования градиента

Выбор величины шага может осуществляться различным образом. Выберем шаг в соответствии с алгоритмом метода скорейшего спуска и получим первое приближение – точку с координатами  $x_1^1, x_2^1$ . Вычисляется значение целевой функции в этой точке  $Z^1$ .

Необходимо проверить, принадлежит ли точка с координатами  $x_1^1, x_2^1$  области  $\Omega$  допустимых значений переменных. Для этого проверяется неравенство (1.23), в которое подставляются координаты  $x_1^1, x_2^1$ :

$$ax_1^1 + bx_2^1 + c \geq 0. \quad (1.23)$$

Если это неравенство выполняется, вычислительный процесс продолжается.

Из точки с координатами  $x_1^1, x_2^1$  выполняется следующий шаг. В результате этого шага имеем второе приближение – точку с координатами  $x_1^2, x_2^2$ . значение целевой функции в этой точке  $Z^2$ .

Пусть для этой точки неравенство  $ax_1^1 + bx_2^1 + c \geq 0$  не выполняется. Следовательно, точка с координатами  $x_1^2, x_2^2$  вышла из области  $\Omega$  и необходимо выполнить возврат в эту область.

Возврат в область  $\Omega$  выполняется следующим образом. Из точки с координатами  $x_1^2, x_2^2$  опускается перпендикуляр на прямую  $ax_1 + bx_2 + c = 0$ , т.е. конец вектора  $(x_1^1, x_2^1; x_1^2, x_2^2)$  проектируется на эту прямую. В результате получается новое приближение – точка с координатами  $x_1^3, x_2^3$ , которая принадлежит области  $\Omega$ . В этой точке вычисляется значение целевой функции  $Z^3$ .

Дальнейший спуск к относительному минимуму целевой функции продолжается из точки  $x_1^3, x_2^3$ . на каждом шаге вычисляется значение целевой функции и проверяется принадлежность нового приближения к области  $\Omega$ . Вычислительный процесс заканчивается при выполнении условия (1.16).

### 1.6 Метод штрафных функций

Рассмотрим задачу поиска локального минимума критерия оптимальности  $W$  в области, ограниченной системой неравенств (3.16)-(3.17). Введение обобщенного критерия оптимальности по методу штрафных функций [3,5] производится с помощью непрерывной функции

$$Q(x) = \sum_{i=1}^m \left[ \frac{l_i(x) + |l_i(x)|}{2} \right]^2. \quad (1.24)$$

Обобщенным критерием оптимальности согласно методу штрафных функций является выражение  $T=W+RQ(x)$ , где  $R$  - некоторое положительное число, называемое коэффициентом штрафа.

Рассматривается некоторая неограниченная, монотонно возрастающая последовательность  $\{R_k\}$ ,  $k=1,2,\dots$  положительных чисел. Для первого элемента этой последовательности с помощью метода покоординатного спуска отыскивается локальный минимум функции  $T$ . Пусть этот минимум достигается при значениях  $(b^*, R_1)$ .

Вектор  $(b^*, R_1)$  используется как начальное приближение для решения задачи поиска минимума функции  $T$  где  $R_2 > R_1$  и т.д. Таким образом, решается последовательность задач минимизации функций  $T(b^*, R_k)$ ,  $k=1,2,\dots$ , причем результат предыдущей оптимизации используется в качестве начального приближения для поиска последующей.

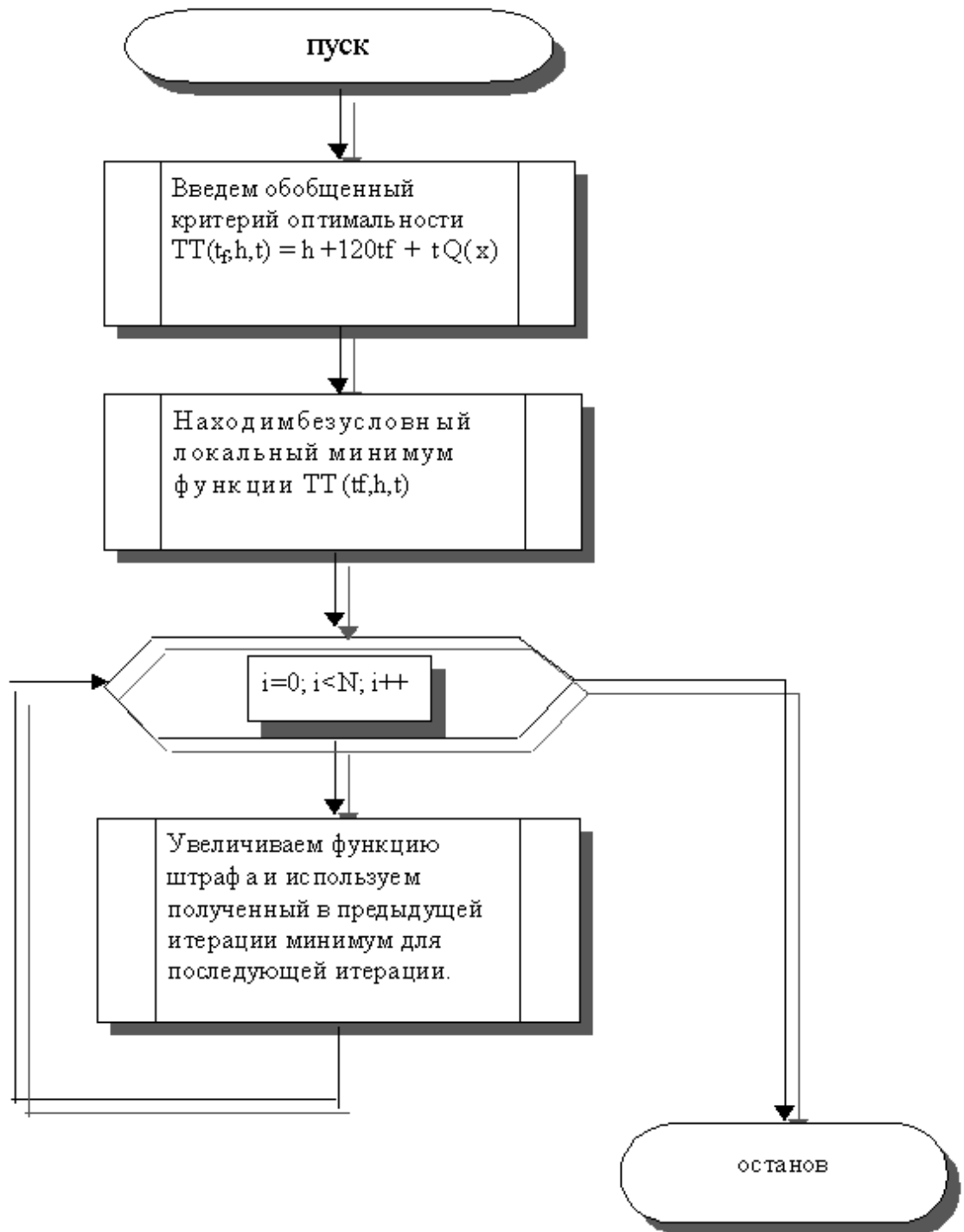


Рисунок 1.5 – Блок-схема метода штрафных функций

### 1.7 Методы полиномиальной аппроксимации

Согласно теореме Вейерштрасса об аппроксимации, непрерывную функцию в некотором интервале можно аппроксимировать полиномом достаточно высокого порядка [6]. Следовательно, если функция унимодальна и найден полином, который достаточно точно ее аппроксимирует, то

координаты точки оптимума функции можно оценить путем вычисления координаты точки оптимума полинома.

Рассмотрим следующие вопросы:

- квадратичная аппроксимация;
- кубическая интерполяция;
- квадратичные функции.

### 1.7.1 Квадратичная аппроксимация

Используется несколько значений функции в определенных точках для аппроксимации функции обычным полиномом по крайней мере в небольшой области значений. Затем положение минимума функции аппроксимируется положением минимума полинома, поскольку последний вычислитель проще.

Простейший случай основан на том факте, что функция, принимающая минимальное значение во внутренней точке интервала, должна быть по крайней мере квадратичной.

Если целевая функция  $W(x)$  в точках  $x_1, x_2, x_3$  принимает соответствующие значения  $W_1, W_2, W_3$ , то можно определить коэффициенты  $a_0, a_1, a_2$  таким образом, что значения квадратичной функции

$$q(x) = a_0 + a_1(x - x_1) + a_2(x - x_1)(x - x_2) \quad (1.25)$$

совпадут со значением  $W(x)$  в трех указанных точках. Вычислим  $q(x)$  в трех указанных точках (см. табл. 1.1).

Таблица 1.1 - Вычисление значений

|                                                                                           |                                                                                                            |
|-------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------|
| $W_1 = W(x_1) = q(x_1) = a_0$                                                             | $a_0 = W_1$                                                                                                |
| $W_2 = W(x_2) = q(x_2) = W_1 + a_1(x_2 - x_1)$                                            | $a_1 = \frac{W_2 - W_1}{x_2 - x_1}$                                                                        |
| $W_3 = W(x_3) = W_1 + \frac{W_2 - W_1}{x_2 - x_1}(x_3 - x_1) + a_2(x_3 - x_1)(x_3 - x_2)$ | $a_2 = \frac{1}{x_3 - x_2} \cdot \left( \frac{W_3 - W_1}{x_3 - x_1} - \frac{W_2 - W_1}{x_2 - x_1} \right)$ |
| $\frac{dq}{dx} = 0$                                                                       | $x^* = \frac{x_2 + x_1}{2} - \frac{a_1}{2a_2}$                                                             |

### 1.7.1.1 Метод Пауэлла

Если известны значения функции  $f(x)$  в трех разных точках  $\alpha, \beta, \gamma$ , равные соответственно  $f_\alpha, f_\beta, f_\gamma$ , то функция  $f(x)$  может быть аппроксимирована квадратичной функцией

$$\varphi(x) = Ax^2 + Bx + C, \quad (1.26)$$

где  $A, B$  и  $C$  определяются из уравнений

$$\begin{aligned} A\alpha^2 + B\alpha + C &= f_\alpha, \\ A\beta^2 + B\beta + C &= f_\beta, \\ A\gamma^2 + B\gamma + C &= f_\gamma. \end{aligned} \quad (1.27)$$

После преобразований этих уравнений получаем

$$\begin{aligned} A &= [(\gamma - \beta)f_\alpha + (\alpha - \gamma)f_\beta + (\beta - \alpha)f_\gamma] / \Delta, \\ B &= [(\beta^2 - \gamma^2)f_\alpha + (\gamma^2 - \alpha^2)f_\beta + (\alpha^2 - \beta^2)f_\gamma] / \Delta, \\ C &= [\beta\gamma(\gamma - \beta)f_\alpha + \gamma\alpha(\alpha - \gamma)f_\beta + \alpha\beta(\beta - \gamma)f_\gamma] / \Delta, \end{aligned} \quad (1.28)$$

где  $\Delta = (\alpha - \beta)(\beta - \gamma)(\gamma - \alpha)$ .

Ясно, что  $\varphi(x)$  будет иметь минимум в точке  $x = -B/2A$ , если  $A > 0$ . Следовательно, можно аппроксимировать точку минимума функции  $f(x)$  значением

$$\delta = \frac{1}{2} \frac{\gamma^2 - \alpha^2}{\gamma - \alpha} \frac{f_\beta + \beta^2 - \gamma^2}{f_\beta + \beta - \gamma} \frac{f_\alpha + \alpha^2 - \beta^2}{f_\alpha + \alpha - \beta} \frac{f_\gamma}{f_\gamma}, \quad (1.29)$$

Этот метод может непосредственно применяться к функциям одной переменной. Он может быть очень полезен для выполнения линейного поиска в процедурах, описанных в теме №3. В этих процедурах требуется найти минимум функции  $f(x)$  в точках прямой  $x_0 + \lambda d$ , где  $x_0$  - заданная точка, а  $d$  определяет заданное направление. Значение функции  $f(x_0 + \lambda d)$  на этой прямой является значениями функции одной переменной  $\lambda$ :

$$\varphi(\lambda) = f(x_0 + \lambda d). \quad (1.30)$$

Идеи и результаты, изложенные выше, преобразуются в вычислительные процедуры, описанные далее. Предположим, что заданы унимодальная функция одной переменной  $f(x)$ , начальная аппроксимация положения минимума и длина шага  $D$ , является величиной того же порядка, что и расстояние от точки  $A$  до точки истинного минимума  $x^*$  (условие, которое не всегда просто удовлетворить). Вычислительная процедура имеет следующие шаги:

Шаг 1.  $x_2 = x_1 + D x$ .

Шаг 2. Вычислить  $W(x_1)$  и  $W(x_2)$ .

Шаг 3.

– если  $W(x_1) > W(x_2)$ , то  $x_3 = x_1 + 2 D x$ ;

– если  $W(x_1) < W(x_2)$ , то  $x_3 = x_1 - D x$ ;  
 $W(x_1) > W(x_2)$ ,

Шаг 4. Вычислить  $W(x_3)$  и найти

$W_{\min} = \min\{W(x_1), W(x_2), W(x_3)\}$ ,

$X_{\min} = x_i$ , соответствующая  $W_{\min}$ .

Шаг 5. По  $x_1, x_2, x_3$  вычислить  $x^*$ , используя формулу для оценивания с помощью квадратической аппроксимации.

Шаг 6. Проверка окончания

– если  $|W_{\min} - W(x^*)| < W$ , то закончить поиск. В противном случае к шагу 7;

– если  $|X_{\min} - x^*| < x$ , то закончить поиск. В противном случае к шагу 7.

Шаг 7. Выбрать  $X_{\min}$  или  $x^*$  и две точки по обе стороны от нее. Обозначить в естественном порядке и перейти к шагу 4.

Заметим, что если точка  $E$  задана слишком малой, то  $\alpha, \beta, \gamma$ , а также  $f_\alpha, f_\beta, f_\gamma$  будут очень близко друг к другу и значение  $\delta$  (1.29) может стать вообще недостижимыми. Чтобы преодолеть эту трудность, перепишем (1.29) для второй и последней интерполяции:

$$\delta = \frac{1}{2} \alpha + \beta + \frac{\frac{1}{2} f_\alpha - f_\beta}{\beta - \gamma} \frac{\beta - \gamma}{f_\alpha + f_\beta + f_\gamma} \frac{\gamma - \alpha}{\alpha - \beta} \quad (1.31)$$

### 1.7.2 Кубическая интерполяция

Квадратичная интерполяция, которая рассматривалась в предыдущем разделе, называется методом Пауэлла и аппроксимирует функцию квадратным трехчленом. В этом разделе рассматривается метод Давидона [6,7], который гарантирует большую точность и аппроксимирует функцию кубическим полиномом.

Для кубической интерполяции в этом методе используются значения функции и ее производной, вычисленных в двух точках.

Рассмотрим задачу минимизации функции  $f(x)$  на прямой  $x_0 + hd$ , то есть минимизацию функции

$$\varphi(h) = f(x_0 + hd) = f(x_0 + hd_1, x_0 + hd_2, \dots, x_0 + hd_n), \quad (1.32)$$

$$\frac{d\varphi}{dh} = \nabla f(x_0 + hd)^T d = g(x_0 + hd)^T d.$$

Предположим, что известные следующие значения:

$$\begin{aligned}
\varphi_p &= \varphi_p, \\
\varphi_q &= \varphi_q, \\
\frac{d\varphi}{dh} p &= G_p, \\
\frac{d\varphi}{dh} q &= G_q.
\end{aligned} \tag{1.33}$$

Эту информацию можно использовать для построения кубического полинома  $a+bh+ch^2+dh^3$ , который будет аппроксимировать функцию  $\varphi(h)$ . Если  $p=0$ , то уравнения, которые определяют  $a, b, c, d$  имеют вид:

$$\begin{aligned}
a &= \varphi_p, \\
a + bq + cq^2 + dq^3 &= \varphi_q, \\
b &= G_p, \\
b + 2cq + 3dq^2 &= G_q.
\end{aligned} \tag{1.34}$$

Следовательно, если  $r$  является точкой минимума кубического полинома,

$$\frac{r}{q} = \frac{G_p + z \pm \left[ G_p + z^2 - G_p G_p + G_q + 2z \right]^{1/2}}{G_p + G_q + 2z} = \frac{G_p + z \pm w}{G_p + G_q + 2z}, \tag{1.35}$$

$$\text{где } w = z^2 - G_p G_q^{1/2}.$$

Одно из значений этого выражения является минимумом. Другая производная равна  $2c + 6dh$ .

Если мы выберем положительный знак, то при  $\frac{h}{p} = \frac{G_p + z + w}{G_p + G_q + 2z}$  другая производная будет

$$\begin{aligned}
& -\frac{2 G_p + z}{q} + \frac{2 G_p + G_q + 2z}{q^2} \cdot \frac{q G_p + z + w}{G_p + G_q + 2z} = \\
& = \frac{1}{q} - 2G_p - 2z + 2G_p + 2z + 2w = \frac{2w}{q} > 0,
\end{aligned} \tag{1.36}$$

$$\frac{r}{q} = \frac{G_p + z + w}{G_p + G_q + 2z} = 1 - \frac{G_q + w - z}{G_q - G_p + 2w} = \frac{z + w - G_p}{G_q - G_p + 2w}. \tag{1.37}$$

Выбор точки  $q$  зависит от нас. Если  $G_p > 0$ , то надо выбрать значение  $q$  положительным, то есть сделать шаг в направлении уменьшения функции  $\varphi(h)$ , в другом случае значения  $q$  надо выбирать отрицательным. Значение должно быть таким, чтобы интервал  $(0, )$  включал в себя минимум. Это будет верным, если  $\varphi_q > \varphi_p$  или  $G_p > 0$ .

Если ни одно из этих условий не выполняется, то мы удваиваем значения  $q$ , повторяя это в случае необходимости, пока указанный интервал не будет содержать в себе минимум.

Давидон, Флетчер и Пауэлл предложили выбирать  $q$  следующим путем:

$$q = \min \left( \eta - \frac{2 \varphi_p - \varphi_m}{G_p} \right), \quad (1.38)$$

где  $\varphi_m$  - оценка самого малого значения истинного минимума  $\varphi(h)$ ,  
 $\eta$  - константа, значение которой обычно берут 2 или 1.

### 1.7.3 Квадратичные функции

Квадратичная функция [7,8]

$$F(x) = a + x^T b + \frac{1}{2} x^T G x, \quad (1.39)$$

где  $a$  - константа;

$b$  - постоянный вектор;

$G$  - положительно определенная симметричная матрица - имеет минимум в точке  $x^*$ , где  $x^*$  определяется следующим путем:

$$\nabla F(x^*) = b + G x^*, \quad (1.40)$$

откуда  $x^* = -G^{-1}b$ .

Любую функцию можно аппроксимировать в окрестности точки  $x_0$  функцией

$$\begin{aligned} \varphi(x) = & f(x_0) + (x - x_0)^T \nabla f(x_0) + \\ & + \frac{1}{2} (x - x_0)^T G(x_0) (x - x_0), \end{aligned} \quad (1.41)$$

где  $G(x_0)$  - матрица Гессе, вычисленная в точке  $x_0$ .

Аппроксимацией минимума функции  $f(x)$  может быть минимум функции  $\varphi(x)$ . Если последний находится в точке  $x_m$ , то

$$\nabla f(x_0) + G(x_0)(x_m - x_0) = 0, \quad (1.42)$$

откуда  $x_m = x_0 - G^{-1}(x_0) \nabla f(x_0)$  или

$$x_m = x_0 - G^{-1}(x_0) g(x_0). \quad (1.43)$$

Таким образом точкой  $x_u$  следующей аппроксимации минимума будет:

$$x_{i+1} = x_i - G^{-1} x_i g x_i \quad (1.44)$$

или

$$x_{i+1} = x_i - \lambda_i G^{-1} x_i g x_i, \quad (1.45)$$

где  $\lambda_i$  - длина шага, определяется одномерным поиском в направлении  $G^{-1}(x_i)g(x_i)$ .

### 1.8 Метод Нелдера-Мида

Метод Нелдера-Мида (поиск по деформируемому многограннику) является развитием симплексного метода Спендли, Хекста и Химсворта [7,8]. Множество  $(n+1)$ -й равноудаленной точки в  $n$ -мерном пространстве называется регулярным симплексом. Эта конфигурация рассматривается в методе Спендли, Хекста и Химсворта. Следовательно, в двумерном пространстве симплексом является равносторонний треугольник, а в трехмерном пространстве правильный тетраэдр. Идея метода состоит в сравнении значений функции в  $(n+1)$  вершинах симплекса и перемещении в направлении оптимальной точки с помощью итерационной процедуры. В симплексном методе, предложенном первоначально, регулярный симплекс использовался на каждом этапе. Нелдер и Мид предложили несколько модификаций этого метода, допускающих, чтобы симплексы были неправильными. В результате получился очень надежный метод прямого поиска, являющийся одним из самых эффективных, если  $n \leq 6$ .

В методе Спендли, Хекста и Химсворта симплекс перемещается с помощью трех основных операций: отражения, растяжения и сжатия. Смысл этих операций станет понятным при рассмотрении шагов процедуры.

Шаг 1. Найдем значения функции в вершинах симплекса:

$$f_1=f(x_1), \quad f_2=f(x_2) \dots f_{n+1}=f(x_{n+1}) \quad (1.46)$$

Шаг 2. Найдем наибольшее значение функции  $f_h$ , следующее за наибольшим значением функции  $f_g$ , наименьшее значение функции  $f_l$  и соответствующие им точки  $x_h$ ,  $x_g$  и  $x_l$ .

Шаг 3. Найдем центр тяжести всех точек, за исключением точки  $x_h$ . Пусть центром тяжести будет

$$x_0 = \frac{1}{n} \sum_{i \neq h} x_i \quad (1.47)$$

и вычислим  $f(x_0)=f_0$ .

Шаг 4. Удобнее всего начать перемещение от точки  $x_h$ . Отразив точку  $x_h$  относительно точки  $x_0$ , получим точку  $x_r$  и найдем  $f(x_r) = f_r$ .

Операция отражения иллюстрируется рис. 1.6. Если  $\alpha > 0$  - коэффициент отражения, то положение точки  $x_r$  определяется следующим образом:

$$x_r - x_0 = \alpha (x_0 - x_h), \text{ т.е.}$$

$$x_r = (1 + \alpha)x_0 - \alpha x_h. \quad (1.48)$$

Замечание.  $\alpha = |x_r - x_0| / |x_0 - x_h|$ .

Шаг 5. Сравним значения функций  $f_r$  и  $f_h$ .

Если  $f_r < f_h$ , то мы получили наименьшее значение функции. Направление из точки  $x_0$  в точку  $x_r$  наиболее удобно для перемещения. Таким образом, мы производим растяжение в этом направлении и находим точку  $x_e$  и значение функции  $f_e = f(x_e)$ . Рисунок 1.7. иллюстрирует операцию растяжения симплекса. Коэффициент растяжения  $\gamma > 1$  можно найти из следующих соотношений:  $x_e - x_0 = \gamma (x_r - x_0)$ , т.е.

$$x_e = \gamma x_r + (1 - \gamma)x_0. \quad (1.49)$$

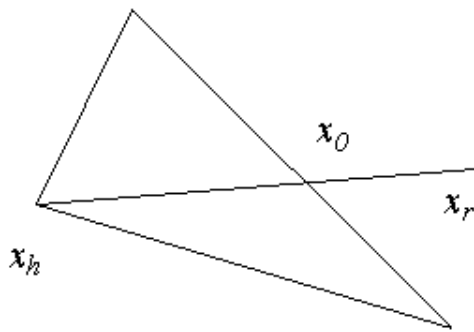


Рисунок 1.6 – Операция отражения

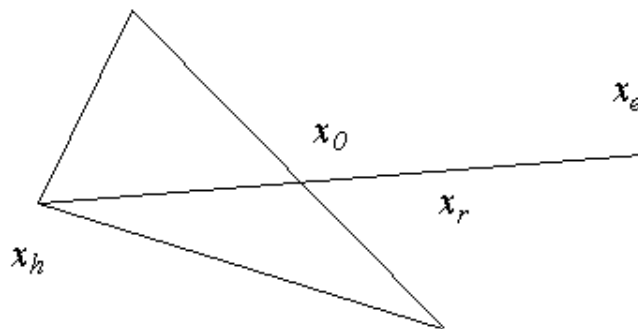


Рисунок 1.7 – Операция растяжения

Замечание.  $\gamma = |x_e - x_0| / |x_r - x_0|$ .

Если  $f_e < f_h$ , то заменяем точку  $x_h$  на точку  $x_e$  и проверяем  $(n+1)$ -ую точку симплекса на сходимость к минимуму (см. шаг 8). Если сходимость достигнута, то процесс останавливается; в противном случае возвращаемся на шаг 2.

Если  $f_e = f_h$ , то отбрасываем точку  $x_e$ . Очевидно, мы переместились слишком далеко от точки  $x_0$  к точке  $x_r$ . Поэтому следует заменить точку  $x_h$  на

точку  $x_r$ , в которой было получено улучшение (шаг 5, 1) проверить сходимость и, если она достигнута, вернуться на шаг 2.

Если  $f_r > f_h$ , но  $f_r \leq f_g$  то  $x_r$  является лучшей точкой по сравнению с другими двумя точками симплекса и мы заменяем точку  $x_h$  на точку  $x_r$  и, если сходимость не достигнута, возвращаемся на шаг 2, т.е. выполняем пункт 1,б, описанный выше.

Если  $f_r > f_h$  и  $f_r > f_g$  то перейдем на шаг 6.

Шаг 6. Сравним значения функций  $f_r$  и  $f_h$ .

Если  $f_r > f_h$ , то переходим непосредственно к шагу сжатия 6,2.

Если  $f_r < f_h$ , то заменяем точку  $x_h$  на точку  $x_r$  и значение функции  $f_h$  на значение функции  $f_r$ . Запоминаем значение  $f_r > f_g$  из шага 5,2, приведенного выше. Затем переходим на шаг 6,2.

В этом случае  $f_r > f_h$ , поэтому ясно, что мы переместились слишком далеко от точки  $x_h$  к точке  $x_0$ . Пытаемся исправить это, найдя точку  $x_c$  (а затем  $f_c$ ) с помощью шага сжатия, показанного на рис. 1.8.

Если  $f_r > f_h$ , то сразу переходим к шагу сжатия и находим точку  $x_c$  из соотношения

$$x_c - x_0 = \beta(x_h - x_0), \quad (1.50)$$

где  $\beta(0 < \beta < 1)$ - коэффициент сжатия. Тогда

$$x_c = \beta x_h + (1 - \beta)x_0. \quad (1.51)$$

Если  $f_r < f_h$ , то сначала заменим точку  $x_h$  на точку  $x_r$ , а затем произведем сжатие. Тогда точку  $x_c$  найдем из соотношения

$$x_c - x_0 = \beta(x_r - x_0), \text{ т.е.}$$

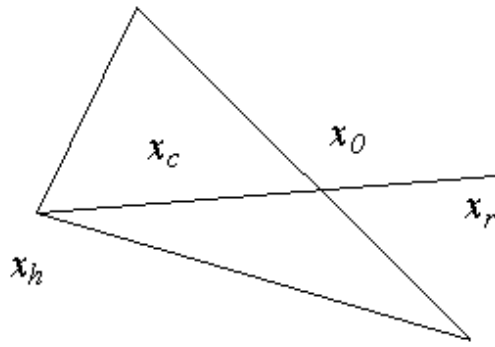
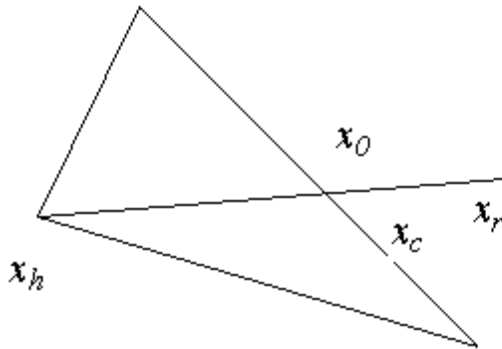
$$x_c = \beta x_r + (1 - \beta)x_0. \quad (1.52)$$

Шаг 7. Сравниваем значения функций  $f_c$  и  $f_h$ .

Если  $f_c < f_h$ , то заменяем точку  $x_h$  на точку  $x_c$ , и если сходимость не достигнута, то возвращаемся на шаг 2.

Если  $f_c > f_h$ , то очевидно, что все наши попытки найти значение меньше  $f_h$  закончились неудачей, поэтому мы переходим на шаг 8.

На этом шаге мы уменьшаем размерность симплекса делением пополам расстояния от каждой точки симплекса до  $x_r$ -точки, определяющей наименьшее значение функции.

Рисунок 1.8 - Шаг сжатия для  $f_r > f_h$ Рисунок 1.9 - Шаг сжатия для  $f_r < f_h$ 

Таким образом, точка  $x_i$  заменяется на точку  $x_i + \frac{1}{2} x_i - x_1$ , т.е.

заменяем точку  $x_i$  точкой  $\frac{1}{2} x_i - x_1$ .

Затем вычисляем  $f_i$  для  $i=1,2,\dots,(n+1)$ , проверяем сходимость и, если она достигнута, возвращаемся на шаг 2.

Шаг 9. Проверка сходимости основана на том, чтобы стандартное отклонение  $(n+1)$ -го значения функции было меньше некоторого заданного малого значения. В этом случае вычисляется

$$\sigma^2 = \sum_{i=1}^{n+1} \frac{f_i - \bar{f}}{n+1}^2, \quad (1.53)$$

$$\text{где } \bar{f} = \sum \frac{f_i}{n+1}.$$

Если  $\sigma < \varepsilon$ , то все значения функции очень близки друг к другу, и поэтому они, возможно, лежат вблизи точки минимума функции  $x_1$ . Исходя из этого, такой критерий сходимости является разумным, хотя Бокс, Дэвис и Свенн предлагают то, что они считают более "безопасной" проверкой.

Коэффициенты  $\alpha, \beta, \gamma$  в вышеприведенной процедуре являются соответственно коэффициентами отражения, сжатия и растяжения. Нелдер и Мид рекомендуют брать  $\alpha=1$ ,  $\beta=0.5$  и  $\gamma=2$ . Рекомендация основана на



$$L = Z(x_1, x_2, \dots, x_n) + \lambda_1 f_1(x_1, x_2, \dots, x_n, b_1) +$$

$$+ \lambda_2 f_2(x_1, x_2, \dots, x_n, b_2) + \dots + \lambda_m f_m(x_1, x_2, \dots, x_n, b_m) \rightarrow \text{extr},$$

где  $\lambda_1, \lambda_2, \dots, \lambda_m$  - неопределенные множители Лагранжа, являющиеся, как и переменные  $x_1, x_2, \dots, x_n$ , искомыми переменными.

Видно, что в функцию Лагранжа входит целевая функция плюс каждое ограничение, умноженное на множитель Лагранжа.

Доказано, что относительный экстремум целевой функции (1.55) при ограничениях (1.57) совпадает с абсолютным экстремумом функции Лагранжа (1.58).

Поиск абсолютного экстремума функции (1.58) выполняется известными методами. В частности, определяются и приравниваются к нулю частные производные функции Лагранжа:

$$\frac{\partial L}{\partial x_1} = \frac{\partial Z}{\partial x_1} + \lambda_1 \frac{\partial f_1}{\partial x_1} + \lambda_2 \frac{\partial f_2}{\partial x_1} + \dots + \lambda_m \frac{\partial f_m}{\partial x_1} = 0,$$

$$\frac{\partial L}{\partial x_2} = \frac{\partial Z}{\partial x_2} + \lambda_1 \frac{\partial f_1}{\partial x_2} + \lambda_2 \frac{\partial f_2}{\partial x_2} + \dots + \lambda_m \frac{\partial f_m}{\partial x_2} = 0,$$

.....

$$\frac{\partial L}{\partial x_n} = \frac{\partial Z}{\partial x_n} + \lambda_1 \frac{\partial f_1}{\partial x_n} + \lambda_2 \frac{\partial f_2}{\partial x_n} + \dots + \lambda_m \frac{\partial f_m}{\partial x_n} = 0,$$

$$\frac{\partial L}{\partial \lambda_1} = f_1(x_1, x_2, \dots, x_n, b_1) = 0,$$

$$\frac{\partial L}{\partial \lambda_2} = f_2(x_1, x_2, \dots, x_n, b_1) = 0,$$

.....

$$\frac{\partial L}{\partial \lambda_m} = f_m(x_1, x_2, \dots, x_n, b_1) = 0.$$

(1.59)

Последние  $m$  уравнений представляют собой ограничения (1.57) оптимизационной задачи.

Система (1.59) содержит  $(m+n)$  уравнений и такое же количество неизвестных.

Решение системы (1.59) даст координаты абсолютного минимума функции Лагранжа (1.58) или относительного минимума целевой функции (1.55) при ограничениях (1.57).

Решение системы (1.59) выполняется известными методами вычислительной математики. Если система (1.59) линейная, используется, как правило, метод Гаусса. Если система (1.59) нелинейная – метод Ньютона.

### ***1.10 Выбор метода оптимизации***

Перед выбором метода оптимизации, проведем краткий анализ задач, которые должно решать разрабатываемое программное обеспечение:

- программа должна решать задачу условной минимизации, т.е. находить относительный экстремум, так как в математической модели кроме линейных ограничений будут иметь место и нелинейные;
- так как целевая функция – функция нескольких переменных, то она может иметь несколько экстремумов, и в этом случае программа должна осуществлять поиск локального минимума.

Проведя анализ наиболее часто использующихся методов оптимизации, для реализации поставленной цели был выбран градиентный метод квадратичного программирования, который представляет собой наиболее эффективный из вышеперечисленных градиентных методов, модифицированный с методами полиномиальной аппроксимации.

Предполагается, что целевая функция и граничные условия аппроксимируются квадратичными зависимостями или полиномами второго порядка. Более подробно этот метод будет рассмотрен далее в разделе "Разработка программного обеспечения метода оптимизации".

Данный метод позволяет создать надежную программу, соответствующую всем вышеперечисленным требованиям.

## **2 Разработка метода оптимизации по реактивной мощности**

Требуемая в электроэнергетической системе (ЭЭС) суммарная мощность компенсирующих устройств определяется из уравнения баланса реактивной мощности (6.1). Эту мощность необходимо разместить в узлах электрической сети с минимальными затратами.

$$\sum Q_{\Gamma} = \sum Q_H + \sum Q_{CH} + \Delta Q_{\Sigma} = \sum Q_{\Pi}, \quad (2.1)$$

где  $\sum Q_{\Gamma}$  - суммарная реактивная мощность, генерируемая в ЭЭС, включая реактивную мощность, поступающую из соседних ЭЭС;

$\sum Q_H$  - суммарная реактивная мощность потребителей ЭЭС, включая реактивную мощность, отдавая в соседние ЭЭС;

$\sum Q_{CH}$  - суммарная реактивная мощность собственных нужд электростанций;

$\Delta Q_{\Sigma}$  - суммарные потери реактивной мощности;

$\sum Q_{\Pi}$  - суммарное потребление реактивной мощности в ЭЭС.

Рассмотрим простейшую схему существующей сети (рис.2.1). от источника питания с напряжением  $U$  через сопротивление сети  $R$  получает питание нагрузка мощностью  $S = P + jQ$  [9]. На шинах нагрузки установлено компенсирующее устройство мощностью  $Q_k$ .

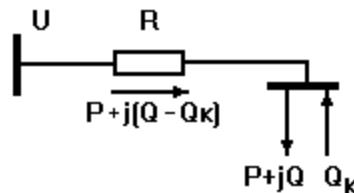


Рисунок 2.1 – Простейшая схема компенсации реактивной мощности

Потери активной мощности в линии при отсутствии у потребителя компенсирующего устройства ( $Q_k = 0$ ) составляют

$$\Delta P = \frac{P^2 + Q^2}{U^2} R. \quad (2.2)$$

При установке у потребителя компенсирующего устройства ( $Q_k \neq 0$ ) эти потери уменьшатся до величины

$$\Delta P = \frac{P^2 + (Q - Q_k)^2}{U^2} R. \quad (2.3)$$

Таким образом, компенсация реактивной мощности позволяет уменьшить потери активной мощности в схеме электроснабжения и, следовательно, улучшить технико-экономические показатели этой схемы.

Оценим влияние КУ на затраты в сети.

Выражение для суммарных затрат на передачу мощности к нагрузке при установке КУ будет иметь вид:

$$Z = Z_K + c_o \Delta P = z_K Q_K + c_o \frac{P^2 + Q - Q_K^2 R}{U^2}, \quad (2.4)$$

где  $Z_K$  – затраты на КУ;

$c_o \Delta P$  – затраты на покрытие потерь активной мощности в сети;

$c_o$  – стоимость единицы потерянной активной мощности;

$z_K$  – удельные затраты на КУ.

Для определения минимума функции  $Z$  приравняем к нулю ее производную от переменной  $Q_K$ :

$$\frac{dZ}{dQ_K} = z_K - \frac{2 Q - Q_K R c_o}{U^2} = 0. \quad (2.5)$$

Из (2.5) определяется экономически целесообразная реактивная мощность, передача которой от источника к потребителю отвечает минимуму затрат  $Z$

$$Q_{\text{э}} = Q - Q_K = \frac{z_K U^2}{2 R c_o}. \quad (2.6)$$

Величина  $Q_{\text{э}}$  не зависит от активной мощности  $P$ , а зависит лишь от соотношения стоимостных показателей  $z_K$  и  $c_o$  и параметров сети  $U$  и  $R$ , по которой передается мощность.

Вопрос о размещении компенсирующих устройств в электрической сети реальной ЭЭС представляет собой сложную оптимизационную задачу. Сложность заключается в том, что электроэнергетические системы являются большими системами, состоящими из взаимосвязанных подсистем. Рассматривать изолированно каждую отдельную подсистему нельзя, поскольку свойства больших систем определяются характером взаимосвязей отдельных подсистем.

При анализе больших систем используется системный подход [9,10,11], согласно которому анализ большой системы выполняется при разделении ее на подсистемы, непосредственно не связанные между собой, но влияющие друг на друга через систему более высокого уровня.

Применительно к рассматриваемому вопросу электрическая сеть представляется разными уровнями, как это показано на рис. 2.2. верхний

уровень – это электрическая сеть напряжением 110 кВ и выше. Эта сложноразветвленная электрическая сеть, представляемая полной схемой замещения, показана на рис.2.2 условно, как ЭС1. Реактивные мощности, вырабатываемые генераторами электростанций  $Q_{ЭС}$ , компенсирующими устройствами  $Q_K$ , линиями электропередачи  $Q_C$ , а также реактивные мощности, протекающие по связям с соседними ЭС2 и ЭС3 ( $Q_{12}$ ,  $Q_{21}$ ,  $Q_{13}$ ,  $Q_{31}$ ) обеспечивают в ЭС1 располагаемую реактивную мощность  $Q_{pl}$ .

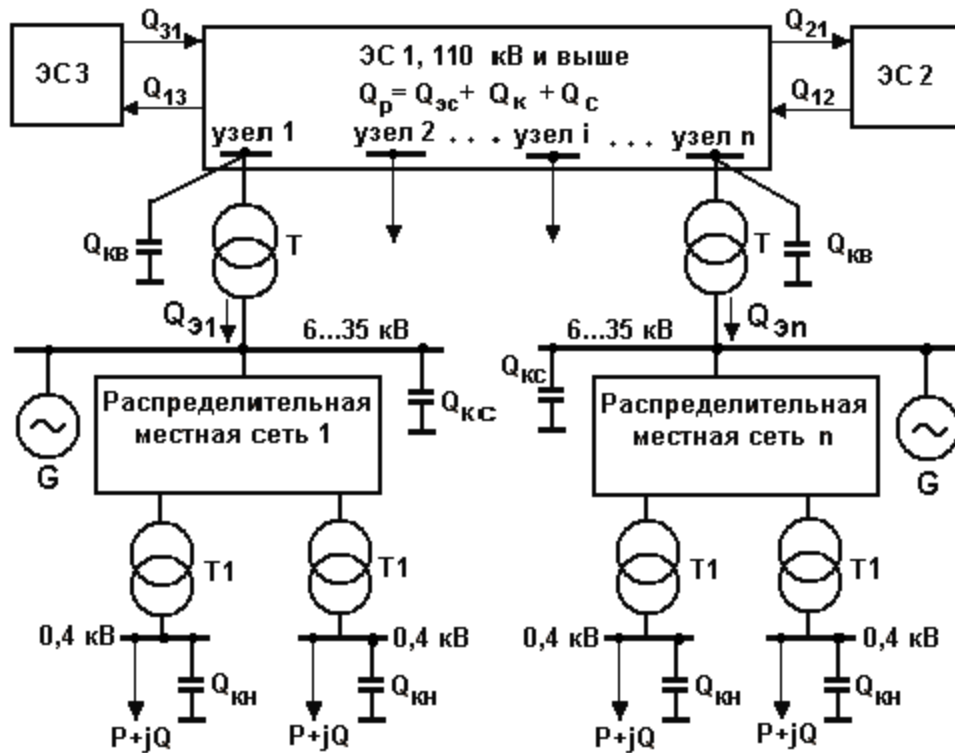


Рисунок 2.2 – Схема размещения КУ в электрической сети

Второй уровень это множество  $n$  разомкнутых местных распределительных сетей напряжением 35 кВ и ниже, присоединенных к  $n$  узлам электрической сети верхнего уровня через трансформаторы Т. Эти местные распределительные сети непосредственно не связаны между собой, но влияют друг на друга через сеть верхнего уровня. Синхронные генераторы, компенсаторы и двигатели в каждой такой распределительной сети представлены одной эквивалентной синхронной машиной G. От местных электрических сетей через распределительные трансформаторы Т1 питаются низковольтные потребители  $P+jQ$ .

Компенсирующие устройства могут устанавливаться на шинах высшего ( $jQ_{кв}$ ) и низшего ( $jQ_{кс}$ ) напряжения трансформаторов Т, а также на шинах 0,4 кВ распределительных трансформаторов Т1 и в самой сети 0,4 кВ ( $jQ_{кн}$ ). Значение мощностей этих КУ и подлежит определению.

В общем виде задача оптимизации размещения КУ формулируется следующим образом: определить реактивные мощности имеющихся в узлах 6...35 кВ синхронных машин  $G$ , мощности КУ в сетях всех напряжений  $Q_{кв}$ ,  $Q_{кс}$ ,  $Q_{кн}$ , а также значения реактивных мощностей  $Q_{gi}$  ( $i=1, 2, \dots, n$ ), передаваемых в сети потребителей, при которых обеспечивается минимум суммарных затрат.

Расчеты компенсации реактивной мощности для сетей всех видов выполняются как при проектировании развития электрических сетей, так и в условиях их эксплуатации. При проектировании определяются мощности КУ и решается задача их распределения в электрической сети. В условиях эксплуатации определяют оптимальные режимы имеющихся КУ в течение суток. Критериями оптимальности в этом случае служат минимум потерь мощности и энергии и соответствие отклонений напряжений допустимым значениям.

При проектировании схемы электроснабжения, как правило, минимизируются денежные затраты на эту схему. Снижение потерь мощности за счет установки КУ уменьшает затраты на схему, по следующим причинам:

- каждый потерянный кВт мощности необходимо выработать на электростанциях и, следовательно, затратить на это денежные средства;
- генерация недополученной реактивной мощности на электростанциях обходится гораздо дороже, чем потребление (в 3 раза!).

Однако и компенсирующие устройства требуют денежных затрат.

В связи с этим возникает задача определения оптимальной мощности компенсирующих устройств, отвечающей минимуму суммарных затрат. Такая задача относится к задаче безусловной оптимизации и может быть решена, например, градиентными методами.

Рассмотрим такую задачу для магистральной схемы электроснабжения (рис. 2.3). Необходимо определить мощности компенсирующих устройств  $Q_{к1}$  и  $Q_{к2}$  в узлах 1 и 2 исходя из условия минимума суммарных затрат на установку этих устройств и покрытие потерь активной мощности в схеме.

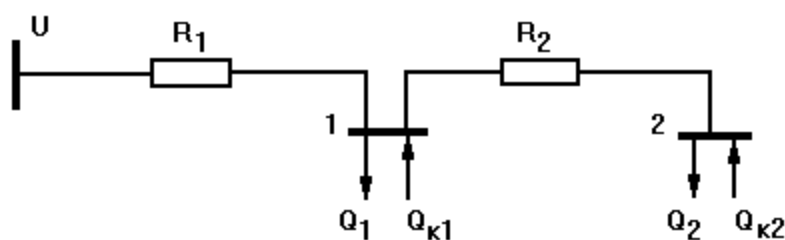


Рисунок 2.3 – Схема электроснабжения

Исходные данные:

напряжение схемы  $U$ ;

сопротивления линий  $R_1$  и  $R_2$ ;

реактивные нагрузки узлов 1 и 2  $Q_1$  и  $Q_2$ ;

удельные затраты на установку компенсирующих устройств  $z_0$ ;

удельные затраты на покрытие потерь активной мощности  $c_0$ .

Целевая функция, представляющая собой суммарные затраты на установку компенсирующих устройств и покрытие потерь активной мощности в схеме, имеет следующий вид

$$Z = z_0 (Q_{k1} + Q_{k2}) + a_1 (Q_1 + Q_2 - Q_{k1} - Q_{k2})^2 + a_2 (Q_2 - Q_{k2})^2 \rightarrow \min, \quad (2.7)$$

где  $a_1 = R_1 \cdot c_0 \cdot 10^{-3} / U^2 = 0,0006$ ;

$a_2 = R_2 \cdot c_0 \cdot 10^{-3} / U^2 = 0,0004$ .

Введение числового коэффициента  $10^{-3}$  необходимо для приведения всех составляющих целевой функции к одной размерности (у.е.).

Для решения задачи выберем метод покоординатного спуска. Определим частные производные целевой функции  $Z$  по переменным  $Q_1$  и  $Q_2$ :

$$\begin{aligned} \frac{\partial Z}{\partial Q_{k1}} &= z_0 - 2a_1 (Q_1 + Q_2 - Q_{k1} - Q_{k2}), \\ \frac{\partial Z}{\partial Q_{k2}} &= z_0 - 2a_1 (Q_1 + Q_2 - Q_{k1} - Q_{k2}) - 2a_2 (Q_2 - Q_{k2}). \end{aligned} \quad (2.8)$$

Примем исходное приближение:

$$\begin{aligned} Q_{k1}^0 &= 0, \\ Q_{k2}^0 &= 0. \end{aligned} \quad (2.9)$$

Для этих значений вычислим значения целевой функции и ее частных производных.

Примем, что в направлении переменной  $Q_{k2}$  целевая функция  $Z$  убывает сильнее, чем в направлении переменной  $Q_{k1}$ , т.е.

$$\left| \frac{\partial Z}{\partial Q_{k2}} \right| > \left| \frac{\partial Z}{\partial Q_{k1}} \right|. \quad (2.10)$$

В направлении переменной  $Q_{k2}$  и начнем спуск.

Примем величину шага  $\lambda = 400$  квар. Первое приближение (первый шаг) будет  $Q_{k1}^1 = 0$ ,  $Q_{k2}^1 = 400$  квар. Рассчитываем значение целевой функции  $Z^1$ .

Второй шаг:  $Q_{k1}^2 = 0$ ,  $Q_{k2}^2 = 400$  квар. Рассчитываем значение целевой функции  $Z^2$ .

Спуск по координате  $Q_{k2}$  следует продолжать до тех пор, пока  $Z^n < Z^{n-1}$ . Как только  $Z^n$  становится больше предыдущего значения  $Z^{n-1}$ , следует спуск по координате  $Q_{k2}$  прекратить и вернуться к значениям переменных  $Q_{k1}^{n-1}$  и  $Q_{k2}^{n-1}$ , полученным на  $n-1$  шаге.

Выполним новый шаг в направлении другой переменной  $Q_{k1}$ . Находится новое значение целевой функции  $Z$ . Спуск по этой переменной продолжается так же, как и в направлении  $Q_{k2}$  – до тех пор, пока  $Z^m < Z^{m-1}$ .

Точка с полученными координатами  $Q_{k1}^{m-1}$ ,  $Q_{k2}^{n-1}$  находится в окрестности минимума целевой функции  $Z$ . При принятой длине шага  $\lambda=400$ квар более точное решение получено быть не может. Для получения более точного решения необходимо уменьшить шаг и продолжить спуск. Абсолютно точно что, чем меньше шаг, тем точнее будет результат. Посредством ручного расчета мы не можем добиться такой точности. Для решения этой задачи целесообразно будет использовать программное обеспечение, предназначенное для решения задачи нелинейного программирования с нелинейными ограничениями. Одним из таких языков программирования является язык C++.

Это была рассмотрена задача безусловной оптимизации, т.е. нахождения абсолютного минимума. При решении поставленной задачи для нахождения оптимального режима работы сети ОАО "ММК им. Ильича" требуется найти относительный минимум, так как система ограничений будет иметь нелинейный вид (см. далее "Разработка программного обеспечения"). Таким образом, перед нами ставится задача условной оптимизации по реактивной мощности, для которой мы применяем выбранный ранее градиентный метод квадратичного программирования.

### ***3.Разработка программного обеспечения метода оптимизации***

Для оптимизации режимов по реактивной мощности разработан комплекс программ (см. рис. 3.1). Его условно можно разделить на две части:

- интерфейсная часть, разработанная в удобном, понятном виде, предназначенная для работы с пользователем любого уровня;
- расчетная часть, которая непосредственно выполняет необходимые расчеты для получения оптимальных режимов.

В интерфейсную часть комплекса входят программы ввода следующих параметров:

- линий;
- трансформаторов;
- компенсирующих устройств;
- реакторов;
- нагрузок.

В расчетную часть входят следующие программы:

- формирования узловой матрицы;
- формирования векторов узловых токов без оптимизации мощностей компенсирующих устройств;
- расчета узловых напряжений;
- формирования векторов ограничений узловых токов;
- расчета оптимальных значений узловых напряжений;
- расчета оптимальных значений мощностей компенсирующих устройств.

Для расчета установившегося неоптимального режима разработаны программы:

- формирования матриц узловых проводимостей,
- формирования узловых токов источников,
- ЛУ разложения матрицы,
- решение систем уравнений узловых напряжений.

Для расчета оптимального режима разработана программа условной оптимизации с нелинейными ограничениями. В качестве исходных данных для этой программы используется узловые напряжения, рассчитанные для неоптимального режима. Оптимизация производится градиентным методом квадратичного программирования.

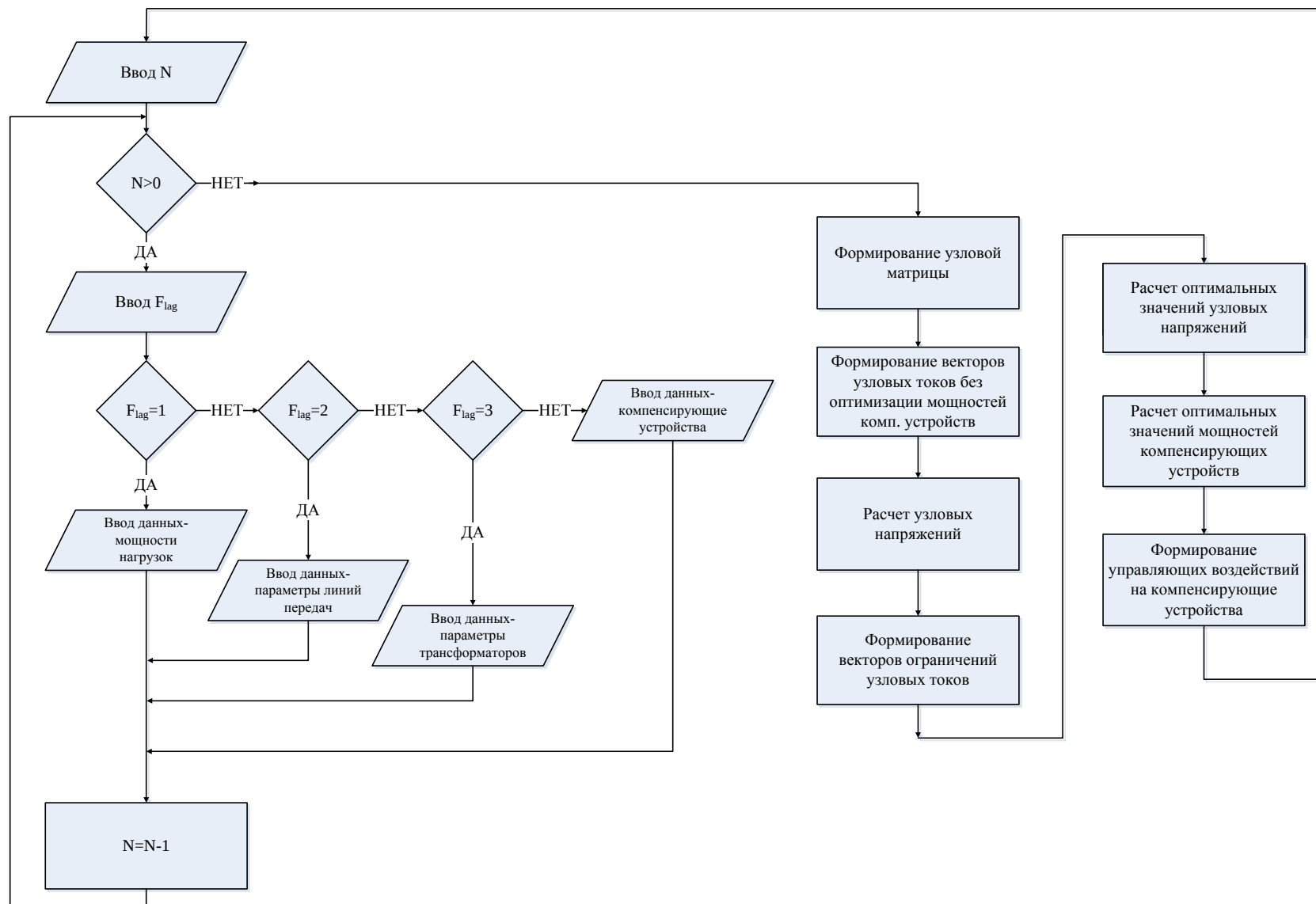


Рисунок 3.1 – Блок-схема разработанного программного обеспечения

Программа расчета неоптимального установившегося режима включает в себя алгоритм треугольного разложения матрицы и итерационный алгоритм решения системы нелинейных уравнений подобный методу Гаусса-Зейделя, модифицированный для решения сетевых нелинейных задач. [2,4,8] Особенностью итерационного алгоритма является то, что на каждой последующей итерации для определения узловых токов источников используется значения узловых напряжений, полученное на предыдущей итерации (см. 3.1).

$$j_{K+1} = \left( \frac{\dot{S}}{\dot{U}_K} \right)^* . \quad (3.1)$$

Для решения задачи нелинейного программирования и определения оптимальных узловых напряжений был разработана программа NCONF CPP, которая реализует последовательный алгоритм квадратичного программирования и конечноразностный градиент. [12]

NCONF CPP (m, me, n, xguess, ibtype, xlb, xub, xscale, iprint, maxitn; x, fvalue).

Параметры программы NCONF CPP:

входные: m, me, n, xguess, ibtype, xscale, iprint, maxitn;

входной/выходной: xlb, xub;

выходные: x, fvalue.

m - общее число ограничений.

me – число ограничений равенства.

n – число переменных.

x – вектор начальных значений напряжений (активных и реактивных составляющих).

xguess – вектор размена n, содержащий начальное предположение о значениях напряжений;

ibtype – скаляр, задающий вид ограничений на переменные:

ibtype=3 – задаем ограничение на напряжение первого узла, все остальные будут иметь те же ограничения.

xlб – вектор размера n, содержащий нижние границы переменных:

- входная, если ibtype=0;
- выходная, если ibtype=1 или 2;
- входная/выходная, если ibtype=3.

xub - вектор размера n, содержащий верхние границы переменных:

- входная, если  $ibtype=0$ ;
- выходная, если  $ibtype=1$  или  $2$ ;
- входная/выходная, если  $ibtype=3$ .

$xscale$  – вектор размера  $n$ , содержащий диагональную матрицу масштабирования переменных. Все элементы вектора  $xscale$  равны  $1.0$ .

$iprint$  – параметр, задающий желаемый уровень печати; принимает следующие значения:

- $0$  – нет печати;
- $1$  – выводится итоговый анализ о работе программы;
- $2$  – дополнительно на каждой итерации выводится одна строка с промежуточными результатами;
- $3$  – о каждой итерации выводится детальная информация.

$maxitn$  – максимально допустимое число итераций.

$x$  – вектор размера  $n$ , содержащий вычисленное решение.

$fvalue$  – скаляр, содержащий значение целевой функции в полученном решении.

Общая задача нелинейного программирования, решаемая данной программой NCONF CPP, состоит в поиске минимума целевой функции

$$\min_{x \in R^n} f(x) \quad (3.2)$$

с ограничениями

$$\begin{aligned} g_j(x) &= 0, j = 1, \dots, m_c, \\ g_j(x) &\geq 0, j = m_c + 1, \dots, m_c, \\ x_i &\leq x \leq x_u, \end{aligned} \quad (3.3)$$

где все функции задачи являются непрерывно дифференцируемыми.

При решении поставленной задачи целевая функция представляет собой потери мощности и имеет вид:

$$\Delta \dot{S} = \Delta P + j\Delta Q = \sum_{K=1}^M \dot{U}_{Ka} + \dot{U}_{Kp}^2 \cdot g_K + jb_K, \quad (3.4)$$

где  $\dot{U}_{Ka}, \dot{U}_{Kp}$  – соответственно активная и реактивная составляющие векторов узловых напряжений;

$g_K, b_K$  – соответственно активная и реактивная составляющие векторов узловых проводимостей.

Роль нелинейных ограничений выполняет система узловых уравнений с источниками реактивной мощности.



Пусть  $d_k$  – решение подзадачи. Тогда новая точка  $x_{k+1}$  определяется в результате линейного поиска:

$$x_{k+1} = x_k + \lambda d_k, \lambda \in (0,1]. \quad (3.9)$$

Новая точка такова, что в ней функция качества имеет наименьшее значение. В качестве функции качества употребляется функция Лагранжа. Если оптимум не достигнут, то матрица  $B_k$  пересчитывается по положительно определенной формуле секущих.

Главная программа NCONF CPP содержит несколько подпрограмм:  
 foryzc – программа формирования матрицы узловых проводимостей;  
 luc – программа используется для разложения матрицы на треугольные сомножители;  
 rluc – программа, которая отвечает за решение системы уравнений.

## ***4 Разработка адаптивной системы управления режимами электропотребления***

### ***4.1 Функции автоматизированной системы***

Сбор, накопление и передача информации, характеризующей режим электропотребления комбината (информация о нагрузках).

Сбор, накопление и передача информации, характеризующей состояние электрической сети (информация об обрывах линий и переводах в ремонт оборудования)

Передача информации с контрольных точек на диспетчерский пункт и обратно.

Обработка полученной информации, расчет режима электропотребления

Автоматическое изменение параметров устройств, регулирующих реактивную мощность, в местах, где это возможно.

### ***4.2 Описание работы системы***

#### ***4.2.1 Ввод системы в работу***

При первоначальном запуске системы в эксплуатацию собирается информация о конфигурации электрической сети.

Собираются следующие параметры:

- схема сети;
- информация о линиях: длина, марка проводов или кабеля, которыми выполнено питание между подстанциями.

После предварительного расчета, для упрощения работы программного обеспечения в базу данных диспетчерского пункта заносятся следующие параметры:

- узловая топологическая матрица согласно схеме замещения электрической сети;
- матрица узловых проводимостей схемы;
- зарядные мощности линий.

Эти данные хранятся в сервере диспетчерской и используются в качестве исходных данных для последующего расчета режима работы сети. Изменяться эти данные могут в случае изменения конфигурации сети при плановых или аварийных выводах оборудования из работы.

#### 4.2.2 Работа системы в нормальном режиме

На первом этапе в условиях нормальной работы в определенный момент времени (например, раз в неделю) происходит сбор информации, характеризующей режим электропотребления.

Собираются следующие параметры:

- нагрузки подстанций;
- мощности питающих систем (в данном случае, подстанций "Заря" и "Азовская").

На большинстве подстанций установлено оборудование, которое может быть использовано для учета электроэнергии, например, микропроцессорная релейная защита.

Вторым этапом работы автоматизированной системы является передача по линиям связи собранной информации с подстанций на диспетчерский пункт.

Третьим этапом является расчет текущего режима работы электрической сети, нахождение оптимального режима работы электрической сети, описанный в предыдущих главах пояснительной записки.

Получаем результаты расчета, необходимые для автоматического внедрения оптимального режима работы сети на практике: мощности компенсирующих устройств.

Следующим этапом является передача результатов расчета по линиям связи обратно на подстанцию.

Заключительным этапом можно считать регулирование компенсирующих устройств в соответствии с данными расчета. Желательно исключить человеческий фактор, для этого на каждой подстанции помимо современной релейной защиты, отвечающей за сбор информации, желательно установить контроллеры, которые будут автоматически регулировать мощность компенсирующих устройств в местах, где это возможно.

Для более точной работы автоматической системы необходимо также предусмотреть обратную связь: после всех регулировок происходит новый сбор информации и перепроверка, что новые установленные параметры соответствуют оптимальному электропотреблению. В случае если что-либо было сделано неправильно, вносится коррекция в исходные данные, и весь алгоритм работы автоматической системы повторяется заново.

### 4.2.3 Работа системы в случае изменения конфигурации сети

В случае изменения конфигурации сети: обрыв линий, вывод в ремонт трансформаторов какой-либо подстанции, поломка какого-либо электрооборудования, которое влияет на процесс энергопотребления, - сбор информации происходит вне рабочего графика, т.е. непосредственно после изменения.

В этом случае собираемые данные:

- схема сети;
- информация о линиях: длина, марка проводов или кабеля, которыми выполнено питание между подстанциями;
- нагрузки подстанций;
- мощности питающих систем (в данном случае, подстанций "Заря" и "Азовская").

Вторым предварительным этапом пересчитывается матрица узловых проводимостей, зарядные мощности линий, формируется новая схема замещения сети.

Далее система работает аналогично работе при нормальном режиме.

В случае если данные изменения конфигурации сети планируются на длительный или постоянный срок, результаты предварительного расчета считаются новыми исходными данными и заносятся в базу данных сервера диспетчерской.

Таблица 4.1 – Сводная таблица собираемой и рассчитываемой информации

| № п/п | Этап работы автоматизированной системы | Входные данные                                                                                                    | Полученные результаты                                                                                                    | Дальнейшее использование результатов                                                        |
|-------|----------------------------------------|-------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------|
| 1     | 2                                      | 3                                                                                                                 | 4                                                                                                                        | 5                                                                                           |
| 1     | Предварительный расчет                 | Схема сети; информация о линиях: длина, марка проводов или кабеля, которыми выполнено питание между подстанциями. | узловая топологическая матрица согласно схеме замещения эл. сети; матрица узловых проводимостей зарядные мощности линий. | Заносятся в базу данных сервера диспетчеризации, изменяются при изменении конфигурации сети |

Продолжение таблицы 4.1

| 1 | 2                                                                                            | 3                                                                                   | 4                                                                                                 | 5                                                         |
|---|----------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------|-----------------------------------------------------------|
| 2 | Первый этап – сбор информации                                                                |                                                                                     | нагрузки подстанций; мощности питающих систем (в данном случае, подстанций, "Заря" и "Азовская"). | Второй этап – передача на диспетчерский пункт             |
| 3 | Третий этап – оптимизация текущего режима на языке программирования C++                      | Результаты предварительного расчета + результаты первого этапа (подробнее см. выше) | Оптимальные потери активной и реактивной мощностей; мощности комп. устройств                      |                                                           |
| 4 | Четвертый этап – передача с диспетчерского пункта                                            | Мощности компенсирующих устройств                                                   |                                                                                                   |                                                           |
| 5 | Обратная связь – проверка соответствия между рассчитанными мощностями КУ и отрегулированными | Входные данные предварительного расчета + первого этапа + мощности КУ               | Подтверждение о соответствии или об сообщении об изменении режима                                 | В случае несоответствия – рекомендации к пересчету режима |

### **4.3 Требования к оборудованию и программному обеспечению**

Технический уровень автоматизации должен соответствовать передовому уровню, достигнутому в мировой практике ко времени изготовления нестандартного оборудования и поставки серийно производимой техники.

При внедрении автоматизированной системы на комбинате должны быть предусмотрены средства эффективного контроля работоспособности оборудования, диагностики датчиков, подсистем всех уровней и системы в

целом, как в автоматическом режиме, так и по инициативе обслуживающего персонала.

Вывод параметров должен производиться автоматически на дисплеи и, по требованию, на печать.

Все оборудование вычислительных комплексов, устанавливаемое на участках производства, должно быть промышленного исполнения и отвечать требованиям условий работы на металлургическом производстве.

Оборудование должно работать надежно, ритмично и безаварийно в течение всего годового фонда времени работы стана. Оно должно быть рассчитано на проведение профилактики без нарушения функционирования автоматизированной системы.

Работа нового оборудования должна быть согласована с действующим оборудованием других систем автоматизации.

Связь между подстанциями и диспетчерским пунктом должна осуществляться с использованием стандартных устройств и протоколов. В качестве инструмента для создания систем сбора, обработки и представления информации в графическом виде должны применяться инструментальные оболочки, соответствующие стандартам МЭК с применением языка, используемого в документообороте комбината.

Каких-либо особых требований к оборудованию, устанавливаемому или уже установленному на подстанциях, не имеется, так как данная система универсальна в отношении источников входных данных. Единственным требованием, предъявляемым к приборам учета электроэнергии, является получение данных в аналоговой или дискретной форме. Таким образом, сбор информации для адаптивной системы может происходить любым из перечисленных далее способов:

- с помощью специально предусмотренного оборудования фирмы Allen-Bradley Power Monitor посредством программного обеспечения RSEnergy;
- с помощью современных микропроцессорных релейных защит, способных не только защищать силовые цепи, но и измерять необходимые параметры;
- с помощью контроллера автоматически регулируемых конденсаторных батарей (см. выбор оборудования);
- с помощью специально устанавливаемых измерительных контроллеров.

## **4.4 Выбор оборудования для адаптивной системы**

### **4.4.1 Учет электроэнергии**

Как уже говорилось ранее, адаптивная система универсальна в отношении способа учета электроэнергии. Если на подстанции имеется уже установленная, например, микропроцессорная защита, то установки никакого дополнительного оборудования не требуется. Если же на подстанции нет ничего, что может производить сбор информации в дискретном либо аналоговом виде, то мной предлагается установка оборудования фирмы Allen-Bradley семейства PowerMonitor и программного обеспечения RSEnergy.

#### **4.4.1.1 Устройство мониторинга PowerMonitor**

PowerMonitor – автономное высокотехнологическое устройство управления и мониторинга электроснабжения со встроенными функциями мониторинга, анализа, архивирования и управления[14].

Функции данного устройства:

а) функция отображения:

- 1) напряжения по фазам;
- 2) токи по фазам;
- 3) коэффициент мощности  $\cos \varphi$ ;
- 4) коэффициент искажений;
- 5) активная, реактивная, полная мощность;
- 6) активная, реактивная энергия;
- 7) частота;
- 8) пиковое значение всех перечисленных параметров;
- 9) активная, реактивная энергия: потребленная и сгенерированная;
- 10) коэффициент несимметрии;
- 11) последовательность чередования фаз;

б) функция архивирования:

- 1) регистрация событий во времени;
- 2) регистрация минимальных, максимальных значений;
- 3) более подробная регистрация отдельных событий;
- 4) счетчик активной и реактивной энергии: потребляемой и генерируемой;

в) функция анализа:

- 1) спектральный анализ и осциллографирование;

- 2) коэффициент линейных искажений тока и напряжения;
- г) функция управления:
  - 1) два управляемых однополюсных резервированных контакта, конфигурируемых пользователем;
  - 2) четыре конфигурируемых дискретных входа;
  - 3) коммутационный интерфейс.

#### **4.4.1.2 Программное обеспечение RSEnergy для контроля электропотребления**

Самым оптимальным соотношением цены и качества является установка на каждую подстанцию программного обеспечения фирмы Rockwell Automation RSEnergy [14]. С помощью RSEnergy можно получить все необходимые данные по энергопотреблению на предприятии из любого места сети, в данном случае из диспетчерского пункта. Можно контролировать все энергопотребление по объединенной в сеть корпорации и управлять им (см. рис. 4.1). Для каждого измеряемого параметра предоставляются как данные реального времени, так и накопленные данные.

Имея прямой доступ к оборудованию Allen-Bradley по контролю за электропитанием и PLC посредством RSLinx, можно:

- измерять пиковую нагрузку;
- рассчитывать затраты на электроэнергию;
- определять режим нагрузок;
- измерять качество энергии;
- суммировать нагрузку по нескольким зданиям предприятия или по всей стране.

Достоинства RSEnergy:

- получение оперативной информации;
- оперативный контроль в режиме реального времени;
- разносторонний анализ полученных данных;
- регистрирование событий.



Рисунок 4.1 – Пример сети с использованием RSEnergyMetrix

#### 4.4.2 Процессор для диспетчерского пункта

Модуль PLC представляет собой модуль процессоров, которые обеспечивают проверенный подход для промышленного управления. Процессоры PLC доступны в разном функциональном исполнении и могут соединяться в различные сети при распределенных процессах и для обеспечения удаленного расположения ввода-вывода.

Модульность ввода-вывода и коммуникационных интерфейсов обеспечивает перестраиваемую и расширяемую систему. Изначально конфигурируется система на число входов/выходов и сетевых связей, которое необходимо. Если в последствии возникнет необходимость в расширении системы, можно добавить модули ввода-вывода или коммуникационные интерфейсы.

Процессоры PLC-5 доступны в диапазоне форсируемых входов-выходов от максимум 512 до максимум 3072 и в диапазоне памяти пользователя от максимум 6К до максимум 100К слов. Все процессоры – способны управлять удаленно расположенными модулями ввода-вывода. Максимальное число удаленных от процессора модулей ввода-вывода колеблется от 5 до 125. Процессор может контролировать/управлять этими модулями ввода-вывода через сети ControlNet, DeviceNet и Remote I/O.

### 4.4.3 Сервер связи

RSLink для Rockwell Automation Networks и Devices представляет собой всеобъемлющее решение вопросов организации связи на предприятии при использовании операционных систем Microsoft Windows NT, Windows 95™ и Windows 98™ [14].

Разработанное как реальное 32-разрядное приложение, RSLink использует все преимущества многозадачных возможностей операционных систем Windows. RSLink может одновременно обслуживать любую комбинацию поддерживаемых приложений посредством одинаковых или разных коммуникационных интерфейсов.

RSLink – это готовый сервер связи, обеспечивающий соединение с устройствами на производстве для различных приложений от Rockwell Software. RSLink способен поддерживать множество программных приложений, обеспечивая одновременную связь с разнообразными устройствами в различных сетях. С помощью RSLink можно передавать информацию, имея прямой доступ к PLC как с подстанций (RSEnergy) на диспетчерский пункт так и обратно.

RSLink предоставляет дружественный графический интерфейс для навигации по сети, включая маршрутизацию по корпоративной сети Ethernet с помощью ControlLogix Gateway для обращения к сетям управления и устройствам на производстве (см. рис. 4.2).

Для организации и эксплуатации сетей, начиная от поддерживаемых унаследованных сетей Allen-Bradley и до более новых, обладающих большими возможностями ControlLogix Gateways, предоставляется полный набор драйверов связи. Также поддерживается такое коммуникационное оборудование, как PCMCIA, последовательный порт и сетевые платы на базе ПК.

RSLink поставляется в виде нескольких различных пакетов, удовлетворяющих разнообразным требованиям к стоимости и функциональным возможностям.

RSLink Lite™ предлагает прямую драйверную связь с программируемыми контроллерами для поддержки многих из программных пакетов, предлагаемых Rockwell Software и Allen-Bradley.

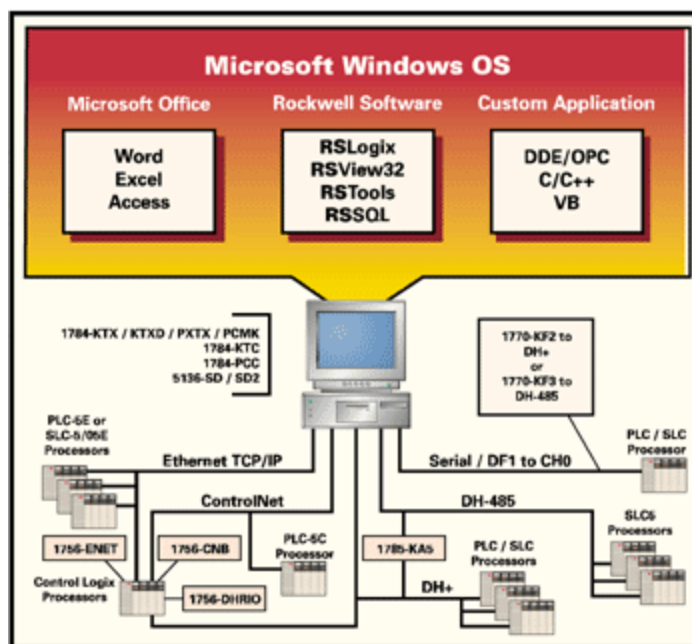


Рисунок 4.2 – Пример организации связи с помощью RSLinx

RSLinx OEM<sup>TM</sup> - коммуникационный процессор для многих программных продуктов MMI, а также для продуктов, разработанных третьей стороной для использования RSLinx C API.

RSLinx<sup>TM</sup> – полная версия RSLinx обеспечивает связь DDE со всеми продуктами, которые могут выступать в качестве клиентов DDE (через наилучший формат DDE, воспринимаемый клиентом DDE: Advance DDE, CF\_Text, XL\_Table или Fast DDE).

RSLinx C SDK – инструментарий для разработки программного обеспечения, который позволяет пользователю создавать приложения с возможностью доступа к коммуникационным возможностям RSLinx OEM или полного RSLinx посредством обращений к библиотеке Dynamic Link Library (DLL) языка Си.

#### 4.4.3.1 Основные преимущества

Простота адаптации к новым процессорам и сетям, поскольку все 32-разрядные драйверы фирмы Allen-Bradley включены в один пакет.

Поддержка для устройств 1784-KT, 1784-KTX(D), 1784-PCMK, RS-232 DFI, сети Ethernet фирмы Allen-Bradley, удаленных устройств через WINtelligent LINX Gateway, RSEmulate 500 и RSEmulate 5.

Совместимость с продуктами программирования фирм Rockwell Software и Allen-Bradley.

Доступ для чтения и записи данных в PLC-5, PLC-2, PLC-3, PLC-5/250<sup>TM</sup>, MicroLogix 1000 и SLC 500.

Одновременная работа множества устройств связи.

Интуитивный пользовательский интерфейс, протестированный в лаборатории по обеспечению удобства в использовании.

Эффективное использование ресурсов системы и минимальный трафик в сети за счет оптимизированных DDE-считываний.

Повышенная скорость и уменьшенная загруженность сети с поблочным считыванием и записью.

Возможность подключения к старым сетям, поддерживаемая удаленной маршрутизацией через 1785-КА, 1785-КА5, 5130-КА и 5820-EI.

Интуитивное обследование системы, поддерживаемое графической функцией SuperWho и комплексной диагностикой.

Помощь, которую можно получить из контекстно-зависимой системы подсказок, просто щелкнув мышью.

#### **4.4.3.2 Минимальные требования RSLinx:**

IBM-совместимый компьютер 468SX или выше.

Требования к RAM будут меняться в зависимости от применения (12 Мбайт минимум).

8 Мбайт свободного пространства на жестком диске.

Графический адаптер и монитор с разрешением VGA или выше.

Версия Microsoft Windows NT<sup>TM</sup> 3.51 или более поздняя (Intel<sup>TM</sup> 486 или Pentium<sup>TM</sup>).

Любое координатно-указательное устройство, совместимое с Windows NT<sup>TM</sup> 3.51 (мышь, шаровой манипулятор, сенсорный экран и т.п.)

Карта Ethernet и/или устройство связи фирмы Allen-Bradley, или кабель, в зависимости от применения.

#### **4.4.3.3 Различия между разными версиями программного обеспечения RSLinx**

Для удовлетворения различных стоимостных и функциональных требований множества различных приложений имеется пять версий программного обеспечения RSLinx. В зависимости от того, с какой версией вы работаете, некоторые функциональные возможности могут быть доступны, а могут быть и недоступны. В разделах, следующих ниже, описываются основные различия между имеющимися версиями.

#### **4.4.3.4 Версия программного обеспечения RSLinx Lite**

RSLinx Lite предоставляет минимум функциональных возможностей, необходимых для поддержки отдельных приложений фирм Rockwell Software и Allen-Bradley. Эта версия не является самостоятельным коммерческим продуктом, а представляет собой составляющую часть программных продуктов, для которых требуется только прямой доступ к сетевым драйверам RSLinx. Эта версия не поддерживает динамический обмен данными (DDE) или выпущенный в обращение интерфейс программирования приложений RSLinx (RSLinx Application Programming Interface (API)). Она не поддерживает возможности связи для многих программных продуктов фирмы Rockwell Software, предназначенных для программирования программируемых контроллеров и отдельных средств автоматизации фирмы Allen-Bradley.

RSLinx Lite обладает следующими функциональными возможностями и преимуществами:

- возможна простая модификация под новые процессоры и сети, поскольку все 32-разрядные драйверы Allen-Bradley включены в один пакет;
- совместимость с продуктами Rockwell Software и Allen-Bradley;
- параллельная работа нескольких коммуникационных устройств;
- возможность связи с имеющимися старыми сетями, обеспечиваемая удаленной маршрутизацией через шесть различных шлюзовых устройств;
- интуитивно-понятная система отслеживания, обеспечиваемая древовидной графической функцией управления RSWho и исчерпывающей диагностикой;
- интуитивно-понятный пользовательский интерфейс, протестированный в лаборатории по изучению вопросов использования программного обеспечения фирмы Rockwell Software;
- получение подсказки по одному щелчку клавишей мыши, при помощи контекстно-зависимой справочной системы.

#### **4.4.3.5 Версия программного обеспечения RSLinx OEM**

RSLinx OEM включает в себя функциональные средства, требующиеся для обслуживания связи со всеми программными продуктами Rockwell Software, предназначенными для работы с программируемыми контроллерами, включая HMI (человеко-машинный интерфейс).

Эта версия включает в себя все функциональные возможности RSLinx Lite, плюс несколько ограниченный AdvanceDDE, предназначенный для работы с программными продуктами Rockwell Software. Она также поддерживает приложения, разработанные для RSLinx Application Programming Interface (API).

Коммуникационные возможности RSLinx позволяют осуществлять связь посредством OPC. OPC основан на технологии OLE фирмы Microsoft и предоставляет стандартный для промышленности механизм связи и обмена данными между клиентами и серверами.

Дополнительно к функциональным возможностям и преимуществам RSLinx Lite, RSLinx OEM позволяет:

- обеспечить совместимость с программными продуктами, разработанными другими производителями или пользователем с использованием API или AdvanceDDE, доступных в RSLinx;
- получить синхронный и асинхронный доступ чтения/записи к процессорным данным в PLC-2®, PLC-3®, PLC-5®, PLC-5/250®, SLC 500™, MicroLogix 1000™, AdvanceDDE и ControlLogix 5550™ через API;
- повысить скорость и уменьшить загрузку сети за счет поблочного чтения/записи;
- повысить эффективность использования ресурсов сети и минимизировать трафик за счет оптимизированных считываний DDE;
- устанавливать «горячую» связь при помощи операции копирования в буфер обмена и извлечения из него для DDE клиентов Rockwell Software;
- устанавливать OPC связь для локальных клиентов;
- поддерживать работу приложений, разработанных для RSLinx API;
- поддерживать AdvanceDDE для программных продуктов Rockwell Software (DDE не поддерживается для DDE клиентов, отличных от Rockwell Software);
- осуществлять диагностику DDE сервера и регистрацию событий.

#### **4.4.3.6 Версия программного обеспечения RSLinx Professional**

RSLinx Professional for Rockwell Automation Networks and Devices – это полнофункциональный программный продукт. RSLinx Professional включает в себя все функции предоставляемые RSLinx Lite и RSLinx OEM, плюс динамический обмен данными (DDE) со всеми DDE клиентами, включая Microsoft Excel и Access, плюс обмен в формате AdvanceDDE программных

продуктов фирмы Rockwell Software со всеми клиентами, понимающими AdvanceDDE.

Дополнительно к функциональным возможностям и преимуществам RSInx Lite и RSInx OEM, полнофункциональная версия RSInx Professional предоставляет следующие возможности:

- возможность простой модернизации версии RSInx Lite и/или RSInx OEM до Linx;
- синхронного и асинхронного доступа чтения/записи к процессорным данным в PLC-2, PLC-3, PLC-5, PLC-5/250, SLC 500, MicroLogix 1000, AdvanceDDE и Logix 5550 через DDE интерфейс;
- стандартный DDE обмен на основе CF\_Text для стандартных DDE клиентов, таких как Visual Basic® и Lotus 1-2-3™;
- стандартный DDE обмен на основе Microsoft XL\_Table с DDE клиентами Microsoft Office, такими как Microsoft Excel и Access;
- обмен между FastDDE и DDE клиентами, понимающими FastDDE;
- OPC связь для локальных клиентов.

#### **4.4.3.7 Версия программного обеспечения RSInx Gateway**

RSInx Gateway for Rockwell Automation Networks and Devices расширяет в рамках предприятия связи на основе RSInx. Клиенты RSInx и WINtelligent LINX имеют возможность напрямую соединяться с драйверами RSInx Gateway по сетям TCP/IP. RSInx Gateway позволяет клиентам напрямую соединяться в режиме «он-лайн» с процессорами PLC, SLC и MicroLogix фирмы Allen-Bradley, которые подсоединены к сетям, доступным станциям RSInx Gateway. Это позволяет направлять данные из цехов в приложения для вывода на экран, сохранения или построения трендов. Это позволяет также настраивать отдельные параметры или готовые загружаемые рецепты для поддерживаемых устройств с машины-диспетчера.

Дополнительно к функциональным возможностям и преимуществам RSInx, программный продукт RSInx Gateway предоставляет следующие возможности:

- организации в рамках сети предприятия широкого доступа для контроля и регистрации данных, и использования готовых рецептов процессорами;
- прямого соединения со всеми драйверами на всех станциях RSInx Gateway вашего предприятия с любого клиента RSInx или WINtelligent LINX, который имеет TCP/IP связь;

- совместимости с Microsoft RAS (сервером удаленного доступа), который позволяет использовать телефонную связь с подразделениями;
- OPC связи с локальными или удаленными клиентами.

#### **4.4.3.8 Версия программного обеспечения RSLinx SDK**

RSLinx SDK for Rockwell Automation Networks and Devices является инструментом для создания пользовательских приложений, которые могут использовать все коммуникационные возможности программных продуктов RSLinx и RSLinx OEM. RSLinx SDK содержит инструментальные файлы, копию RSLinx OEM и примеры программ, словом все, что может понадобиться для создания приложений, совместимых с продуктами фирм Rockwell Software и Allen-Bradley, также использующих RSLinx.

RSLinx SDK обладает следующими функциональными возможностями и преимуществами:

- обладает мощным набором функций доступа к данным и преобразования данных, а также функциями, созданными под многодрайверную архитектуру линии программных продуктов RSLinx;
- ускоряет разработку приложений и ограждает ваши приложения от необходимости детального учета особенностей сетевых протоколов и устройств;
- гарантирует вашим приложениям высокую эффективность и надежную передачу данных;
- позволяет вашим приложениям совместно использовать коммуникационные драйверы со средствами программирования фирмы Rockwell Software и HMI.

#### **4.4.3.9 Графические функции SuperWho и RSWho**

Пункт меню Communications/Super Who выводит на экран все доступные/активные станции коммуникационной сети в окне SuperWho. Основное назначение SuperWho – это показать вам вашу сеть. В режиме Normal (нормальный) или Fast (быстрый) SuperWho представляется в текстовом виде. В графическом (Graphical) режиме SuperWho представляется как графический взгляд на вашу сеть. Вы можете видеть станции в сети, их номера и имена проектов.

Поддерживаются следующие сети: Data Highway Plus, Data Highway 485 и Ethernet. SuperWho действует как инструмент диагностики, когда есть проблемы со связью с определенной станцией.

Используя окно SuperWho, можно вывести на экран счетчики диагностики для любой станции, можно открыть окна счетчиков диагностики для нескольких станций в одной коммуникационной сети.

RSWho - это новый интерфейс просмотра RSlnx, заменивший более ранний интерфейс SuperWho. Он позволяет вам просматривать все активные связи на одном экране (см. рис. 4.3).

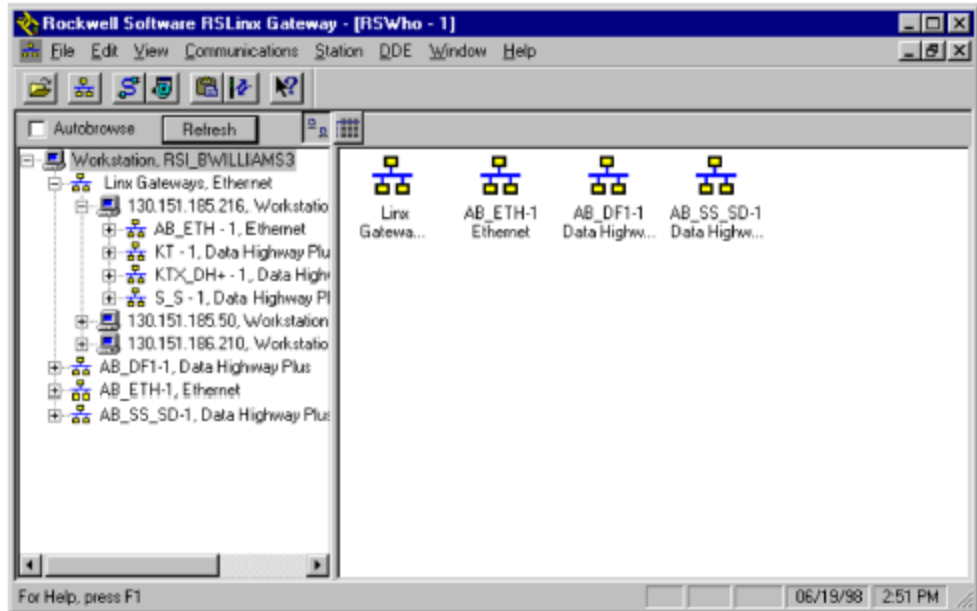


Рисунок 4.3 – Пример окна RSWho

## 5 Исследование и получение оптимальных режимов для ОАО "ММК"

### 5.1 Расчет параметров схемы замещения

#### 5.1.1 Теоретические положения

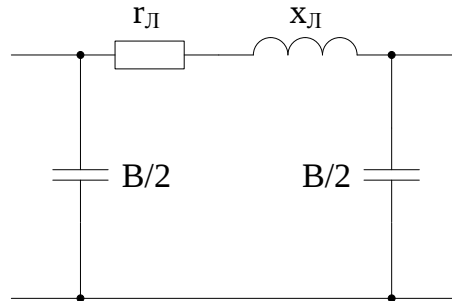


Рисунок 5.1 – Схема замещения линии 110-220 кВ

Активное, реактивное сопротивление линии и емкостная проводимость соответственно рассчитываются по формулам[9,13]:

$$r = r_0 \cdot l, \quad (5.1)$$

$$x = x_0 \cdot l, \quad (5.2)$$

$$b = b_0 \cdot l, \quad (5.3)$$

где  $r_0, x_0, b_0$  - соответственно погонные параметры линий,  
 $l$  - длина линии (кабеля).

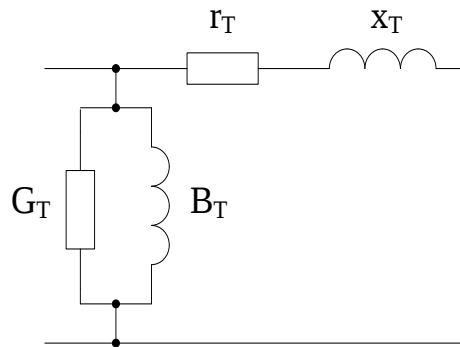


Рисунок 5.2 – Схема замещения двухобмоточного трансформатора

$$r_T = \frac{\Delta P_{\kappa} \cdot U_H^2}{S_H^2}, \quad (5.4)$$

$$x_T = \frac{U_{\kappa\%} \cdot U_H^2}{100 \cdot S_H}, \quad (5.5)$$

$$G_T = \frac{\Delta P_{xx}}{U_H^2}, \quad (5.6)$$

$$B_T = \frac{I_{0\%} \cdot S_H}{100 \cdot U_H^2}. \quad (5.7)$$

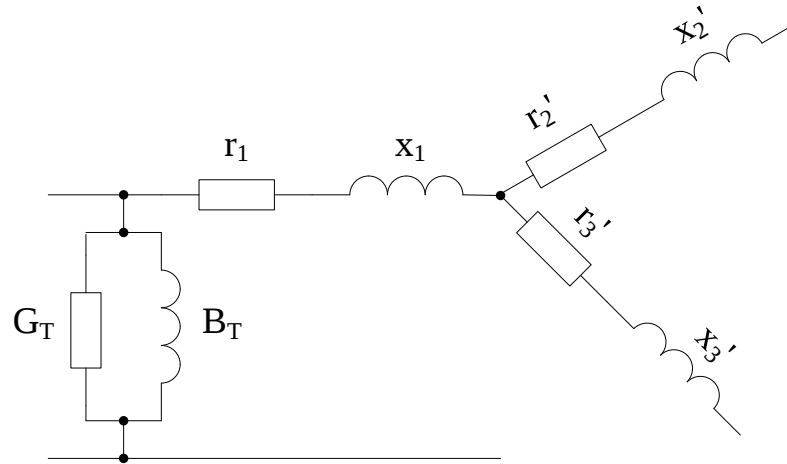


Рисунок 5.3 – Схема замещения трехобмоточного трансформатора  
Расчет активных сопротивлений:

$$r_{1-2} = r_1 + r_2' = \frac{\Delta P_{\kappa 1-2} \cdot U_H^2}{S_H^2}, \quad r_{1-3} = r_1 + r_3' = \frac{\Delta P_{\kappa 1-3} \cdot U_H^2}{S_H^2},$$

$$r_{2-3} = r_2 + r_3' = \frac{\Delta P_{\kappa 2-3} \cdot U_H^2}{S_H^2}.$$
(5.8)

Отсюда

$$r_1 = \frac{r_{12} + r_{13} - r_{23}}{2}, \quad r_2 = \frac{r_{12} + r_{23} - r_{13}}{2},$$

$$r_3 = \frac{r_{13} + r_{23} - r_{12}}{2}.$$
(5.9)

Расчет реактивных сопротивлений:

$$U_{\kappa 1} = 0,5 U_{\kappa 1-2} + U_{\kappa 1-3} - U_{\kappa 2-3},$$

$$U_{\kappa 2} = 0,5 U_{\kappa 1-2} + U_{\kappa 2-3} - U_{\kappa 1-3},$$

$$U_{\kappa 3} = 0,5 U_{\kappa 1-3} + U_{\kappa 2-3} - U_{\kappa 1-2}.$$

$$x_1 = \frac{U_{\kappa 1\%} \cdot U_H^2}{100 \cdot S_H}, \quad x_2 = \frac{U_{\kappa 2\%} \cdot U_H^2}{100 \cdot S_H},$$

$$x_3 = \frac{U_{\kappa 3\%} \cdot U_H^2}{100 \cdot S_H}.$$
(5.11)

Опыт холостого хода проводится аналогично опыту для двухобмоточного трансформатора, поэтому активные и реактивные проводимости в трехобмоточном трансформаторе рассчитываются по тем же формулам (5.7) и (5.8).

### 5.1.2 Расчет параметров схем замещения линий

Кабель АСБГ 4(3\*240) длина 540 м:  $r=70,2 \text{ Ом}$ .

АС-240 длина 604 м:  $r=78,52 \text{ Ом}$ ,  $x=241,6 \text{ Ом}$ ;  $B=1721,4 \cdot 10^{-6} \text{ См}$ .

АС-400 длина 634 м:  $r=50,72 \text{ Ом}$ ,  $x=240,9 \text{ Ом}$ ;  $B=1902 \cdot 10^{-6} \text{ См}$ .

АС-240 длина 686 м:  $r=89,18 \text{ Ом}$ ,  $x=274,4 \text{ Ом}$ ;  $B=1955 \cdot 10^{-6} \text{ См}$ .

### 5.1.3 Расчет параметров схем замещения трансформаторов

Таблица 5.1 – Номинальные параметры трехобмоточного трансформатора п/ст

| Трансформатор  | Среднее<br>номинальное<br>напряжение, кВ | $I_{x.x},$<br>% | $U_{кз}, \%$ |      |     | $\Delta P_{кз},$<br>кВт | $\Delta P_{x.x},$<br>кВт |
|----------------|------------------------------------------|-----------------|--------------|------|-----|-------------------------|--------------------------|
|                |                                          |                 | В-С          | В-Н  | С-Н |                         |                          |
| ТДТН-25000/110 | 115/38,5/6,6                             | 0,7             | 10,5         | 17,5 | 6,5 | 140                     | 31                       |

Согласно формулам (5.9), (5.11) имеем:

$$r_T = 1,5 \text{ Ом},$$

$$x_1 = 56,9 \text{ Ом},$$

$$x_2 = 0,$$

$$x_3 = 35,7 \text{ Ом}.$$

Согласно (5.6)-(5.7):

$$G_T = 0.0023,$$

$$B_T = 0.013.$$

Таблица 5.2 – Номинальные параметры двухобмоточного трансформатора п/ст

| Трансформатор  | Среднее<br>номинальное<br>напряжение, кВ | $U_{кз},$<br>% | $\Delta P_{кз},$<br>кВт | $\Delta P_{x.x},$<br>кВт | $I_{x.x},$ % |
|----------------|------------------------------------------|----------------|-------------------------|--------------------------|--------------|
| ТРДН-40000/110 | 115/6,3                                  | 10,5           | 172                     | 36                       | 0,65         |

Согласно формулам (5.4) - (5.7) имеем:

$$r_T = 1,42 \text{ Ом},$$

$$x_T = 34,7 \text{ Ом},$$

$$G_T = 0.0027,$$

$$B_T = 0.02.$$

Таблица 5.3 – Номинальные параметры двухобмоточного трансформатора Т1 п/ст 28, Т2 п/ст 37

| Трансформатор  | Среднее<br>номинальное<br>напряжение, кВ | $U_{кз},$<br>% | $\Delta P_{кз},$<br>кВт | $\Delta P_{х.х},$<br>кВт | $I_{х.х},$ % |
|----------------|------------------------------------------|----------------|-------------------------|--------------------------|--------------|
| ТДНГ-60000/110 | 115/6,3                                  | 10,5           | 250                     | 50                       | 0,6          |

Подставив номинальные параметры в (5.4) - (5.7) получим:

$$r_T = 0.92 \text{ Ом},$$

$$x_T = 23 \text{ Ом},$$

$$G_T = 0.0038,$$

$$B_T = 0.0272.$$

Таблица 5.4 - Номинальные параметры двухобмоточного трансформатора Т2 п/ст 28, Т1 и Т2 п/ст 20.

| Трансформатор  | Среднее<br>номинальное<br>напряжение, кВ | $U_{кз},$<br>% | $\Delta P_{кз},$<br>кВт | $\Delta P_{х.х},$<br>кВт | $I_{х.х},$ % |
|----------------|------------------------------------------|----------------|-------------------------|--------------------------|--------------|
| ТРДН-63000/110 | 115/6,3                                  | 10,5           | 260                     | 59                       | 0,6          |

Подставив номинальные параметры в (5.4) - (5.7) получим:

$$r_T = 0.87 \text{ Ом},$$

$$x_T = 22 \text{ Ом},$$

$$G_T = 0.0045,$$

$$B_T = 0.0286.$$

Таблица 5.5 - Номинальные параметры двухобмоточного трансформатора п/ст 32

| Трансформатор   | Среднее<br>номинальное<br>напряжение, кВ | $U_{кз},$<br>% | $\Delta P_{кз},$<br>кВт | $\Delta P_{х.х},$<br>кВт | $I_{х.х},$ % |
|-----------------|------------------------------------------|----------------|-------------------------|--------------------------|--------------|
| ТДНГУ-40500/110 | 115/10,5                                 | 10,5           | 172                     | 36                       | 0,65         |

Подставив номинальные параметры в (5.4) - (5.7) получим:

$$r_T = 1,42 \text{ Ом},$$

$$x_T = 34,7 \text{ Ом},$$

$$G_T = 0.0027,$$

$$B_T = 0.0197.$$

Таблица 5.6 - Номинальные параметры двухобмоточного трансформатора Т1 п/ст 33

| Трансформатор | Среднее | $U_{кз},$ | $\Delta P_{кз},$ | $\Delta P_{х.х},$ | $I_{х.х},$ |
|---------------|---------|-----------|------------------|-------------------|------------|
|---------------|---------|-----------|------------------|-------------------|------------|

|                         | номинальное<br>напряжение, кВ | %    | кВт | кВт | %    |
|-------------------------|-------------------------------|------|-----|-----|------|
| Ларкинсон-<br>40500/110 | 115/10,5                      | 10,5 | 172 | 36  | 0,65 |

Подставив номинальные параметры в (5.4) - (5.7) получим:

$$\begin{aligned} r_T &= 1,42 \text{ Ом}, \\ x_T &= 34,7 \text{ Ом}, \\ G_T &= 0.0027, \\ B_T &= 0.0197. \end{aligned}$$

Таблица 5.7 - Номинальные параметры двухобмоточного трансформатора Т2 п/ст 33

| Трансформатор  | Среднее<br>номинальное<br>напряжение, кВ | $U_{кз},$<br>% | $\Delta P_{кз},$<br>кВт | $\Delta P_{х.х},$<br>кВт | $I_{х.х},$<br>% |
|----------------|------------------------------------------|----------------|-------------------------|--------------------------|-----------------|
| ТДНГ-31500/110 | 115/10,5                                 | 10,5           | 140                     | 60                       | 0,7             |

Подставив номинальные параметры в (5.4) - (5.7) получим:

$$\begin{aligned} r_T &= 1,9 \text{ Ом}, \\ x_T &= 44 \text{ Ом}, \\ G_T &= 0.0045, \\ B_T &= 0.0167. \end{aligned}$$

Таблица 5.8 - Номинальные параметры трехобмоточного трансформатора п/ст 6, п/ст 42, Т1 - Т2 п/ст 41

| Трансформатор  | Среднее<br>номинальное<br>напряжение, кВ | $I_{х.х},$<br>% | $U_{кз}, \%$ |     |     | $\Delta P_{кз},$<br>кВт | $\Delta P_{х.х},$<br>кВт |
|----------------|------------------------------------------|-----------------|--------------|-----|-----|-------------------------|--------------------------|
|                |                                          |                 | В-С          | В-Н | С-Н |                         |                          |
| ТДТН-40000/110 | 115/11/6,6                               | 0,6             | 10,5         | 17  | 6   | 200                     | 43                       |

Согласно формулам (5.9), (5.11) имеем:

$$\begin{aligned} r_T &= 0,8 \text{ Ом}, \\ x_1 &= 35,5 \text{ Ом}, \\ x_2 &= 0, \\ x_3 &= 22,3 \text{ Ом}. \end{aligned}$$

Согласно (5.6)-(5.7):

$$\begin{aligned} G_T &= 0.0033, \\ B_T &= 0.0181. \end{aligned}$$

Таблица 5.9 - Номинальные параметры двухобмоточного трансформатора Т1 п/ст 44

| Трансформатор  | Среднее номинальное напряжение, кВ | $U_{кз}, \%$ | $\Delta P_{кз}, \text{кВт}$ | $\Delta P_{х.х}, \text{кВт}$ | $I_{х.х}, \%$ |
|----------------|------------------------------------|--------------|-----------------------------|------------------------------|---------------|
| ТРДН-40000/110 | 115/6,6                            | 10,5         | 172                         | 36                           | 0,65          |

Подставив номинальные параметры в (5.4) - (5.7) получим:

$$r_T = 1,42 \text{ Ом},$$

$$x_T = 34,7 \text{ Ом},$$

$$G_T = 0.0027,$$

$$B_T = 0.0197.$$

Таблица 5.10 - Номинальные параметры двухобмоточного трансформатора Т2 п/ст 44, п/ст 43

| Трансформатор  | Среднее номинальное напряжение, кВ | $U_{кз}, \%$ | $\Delta P_{кз}, \text{кВт}$ | $\Delta P_{х.х}, \text{кВт}$ | $I_{х.х}, \%$ |
|----------------|------------------------------------|--------------|-----------------------------|------------------------------|---------------|
| ТРДН-32000/110 | 115/6,6                            | 10,5         | 140                         | 60                           | 0,7           |

Подставив номинальные параметры в (5.4) - (5.7) получим:

$$r_T = 1,9 \text{ Ом},$$

$$x_T = 44 \text{ Ом},$$

$$G_T = 0.0045,$$

$$B_T = 0.0167.$$

Таблица 5.11 - Номинальные параметры двухобмоточного трансформатора Т1 п/ст 34, п/ст 46

| Трансформатор | Среднее номинальное напряжение, кВ | $U_{кз}, \%$ | $\Delta P_{кз}, \text{кВт}$ | $\Delta P_{х.х}, \text{кВт}$ | $I_{х.х}, \%$ |
|---------------|------------------------------------|--------------|-----------------------------|------------------------------|---------------|
| ТДН-16000/110 | 115/6,6                            | 10,5         | 85                          | 19                           | 0,7           |

Подставив номинальные параметры в (5.4) - (5.7) получим:

$$r_T = 4,38 \text{ Ом},$$

$$x_T = 86,7 \text{ Ом},$$

$$G_T = 0.0014,$$

$$B_T = 0.0085.$$

Таблица 5.12 - Номинальные параметры двухобмоточного трансформатора Т2 п/ст 34, п/ст 2

| Трансформатор | Среднее | $U_{кз},$ | $\Delta P_{кз},$ | $\Delta P_{х.х},$ | $I_{х.х},$ |
|---------------|---------|-----------|------------------|-------------------|------------|
|---------------|---------|-----------|------------------|-------------------|------------|

|                | номинальное<br>напряжение, кВ | %    | кВт | кВт | %   |
|----------------|-------------------------------|------|-----|-----|-----|
| ТДНГ-31500/110 | 115/6,6                       | 10,5 | 140 | 60  | 0,7 |

Подставив номинальные параметры в (5.4) - (5.7) получим:

$$\begin{aligned} r_T &= 1,9 \text{ Ом}, \\ x_T &= 44 \text{ Ом}, \\ G_T &= 0.0045, \\ B_T &= 0.0167. \end{aligned}$$

Таблица 5.13 - Номинальные параметры двухобмоточного трансформатора п/ст 38

| Трансформатор  | Среднее<br>номинальное<br>напряжение, кВ | $U_{кз},$<br>% | $\Delta P_{кз},$<br>кВт | $\Delta P_{х.х},$<br>кВт | $I_{х.х},$<br>% |
|----------------|------------------------------------------|----------------|-------------------------|--------------------------|-----------------|
| ТДНГ-40500/110 | 115/6,6                                  | 10,5           | 172                     | 36                       | 0,65            |

Подставив номинальные параметры в (5.4) - (5.7) получим:

$$\begin{aligned} r_T &= 1,42 \text{ Ом}, \\ x_T &= 34,7 \text{ Ом}, \\ G_T &= 0.0027, \\ B_T &= 0.0197. \end{aligned}$$

Таблица 5.14 - Номинальные параметры двухобмоточного трансформатора Т1 п/ст 37

| Трансформатор   | Среднее<br>номинальное<br>напряжение, кВ | $U_{кз},$<br>% | $\Delta P_{кз},$<br>кВт | $\Delta P_{х.х},$<br>кВт | $I_{х.х},$<br>% |
|-----------------|------------------------------------------|----------------|-------------------------|--------------------------|-----------------|
| ТДНГУ-40500/110 | 115/6,6                                  | 10,5           | 172                     | 36                       | 0,65            |

Подставив номинальные параметры в (5.4) - (5.7) получим:

$$\begin{aligned} r_T &= 1,42 \text{ Ом}, \\ x_T &= 34,7 \text{ Ом}, \\ G_T &= 0.0027, \\ B_T &= 0.0197. \end{aligned}$$

Таблица 5.15 - Номинальные параметры двухобмоточного трансформатора п/ст 40

| Трансформатор | Среднее | $U_{кз},$ | $\Delta P_{кз},$ | $\Delta P_{х.х},$ | $I_{х.х},$ |
|---------------|---------|-----------|------------------|-------------------|------------|
|---------------|---------|-----------|------------------|-------------------|------------|

|               | номинальное<br>напряжение, кВ | %    | кВт | кВт | %   |
|---------------|-------------------------------|------|-----|-----|-----|
| ТДН-31500/110 | 115/6,6                       | 10,5 | 140 | 60  | 0,7 |

Подставив номинальные параметры в (5.4) - (5.7) получим:

$$\begin{aligned} r_T &= 1,9 \text{ Ом}, \\ x_T &= 44 \text{ Ом}, \\ G_T &= 0.0045, \\ B_T &= 0.0167. \end{aligned}$$

Таблица 5.16 - Номинальные параметры двухобмоточного трансформатора п/ст 31: ТДНГ-40500/110/10

| Трансформатор   | Среднее<br>номинальное<br>напряжение, кВ | $U_{кз},$<br>% | $\Delta P_{кз},$<br>кВт | $\Delta P_{х.х},$<br>кВт | $I_{х.х},$<br>% |
|-----------------|------------------------------------------|----------------|-------------------------|--------------------------|-----------------|
| ТДНГУ-40500/110 | 115/11                                   | 10,5           | 172                     | 36                       | 0,65            |

Подставив номинальные параметры в (5.4) - (5.7) получим:

$$\begin{aligned} r_T &= 1,42 \text{ Ом}, \\ x_T &= 34,7 \text{ Ом}, \\ G_T &= 0.0027, \\ B_T &= 0.0197. \end{aligned}$$

Таблица 5.17 - Номинальные параметры двухобмоточного трансформатора ТЗ, Т4 п/ст 41

| Трансформатор   | Среднее<br>номинальное<br>напряжение, кВ | $U_{кз},$<br>% | $\Delta P_{кз},$<br>кВт | $\Delta P_{х.х},$<br>кВт | $I_{х.х},$<br>% |
|-----------------|------------------------------------------|----------------|-------------------------|--------------------------|-----------------|
| ТРДЦН-63000/110 | 115/10                                   | 10,5           | 260                     | 59                       | 0,6             |

Подставив номинальные параметры в (5.4) - (5.7) получим:

$$\begin{aligned} r_T &= 0,87 \text{ Ом}, \\ x_T &= 22 \text{ Ом}, \\ G_T &= 0.0045, \\ B_T &= 0.0286. \end{aligned}$$

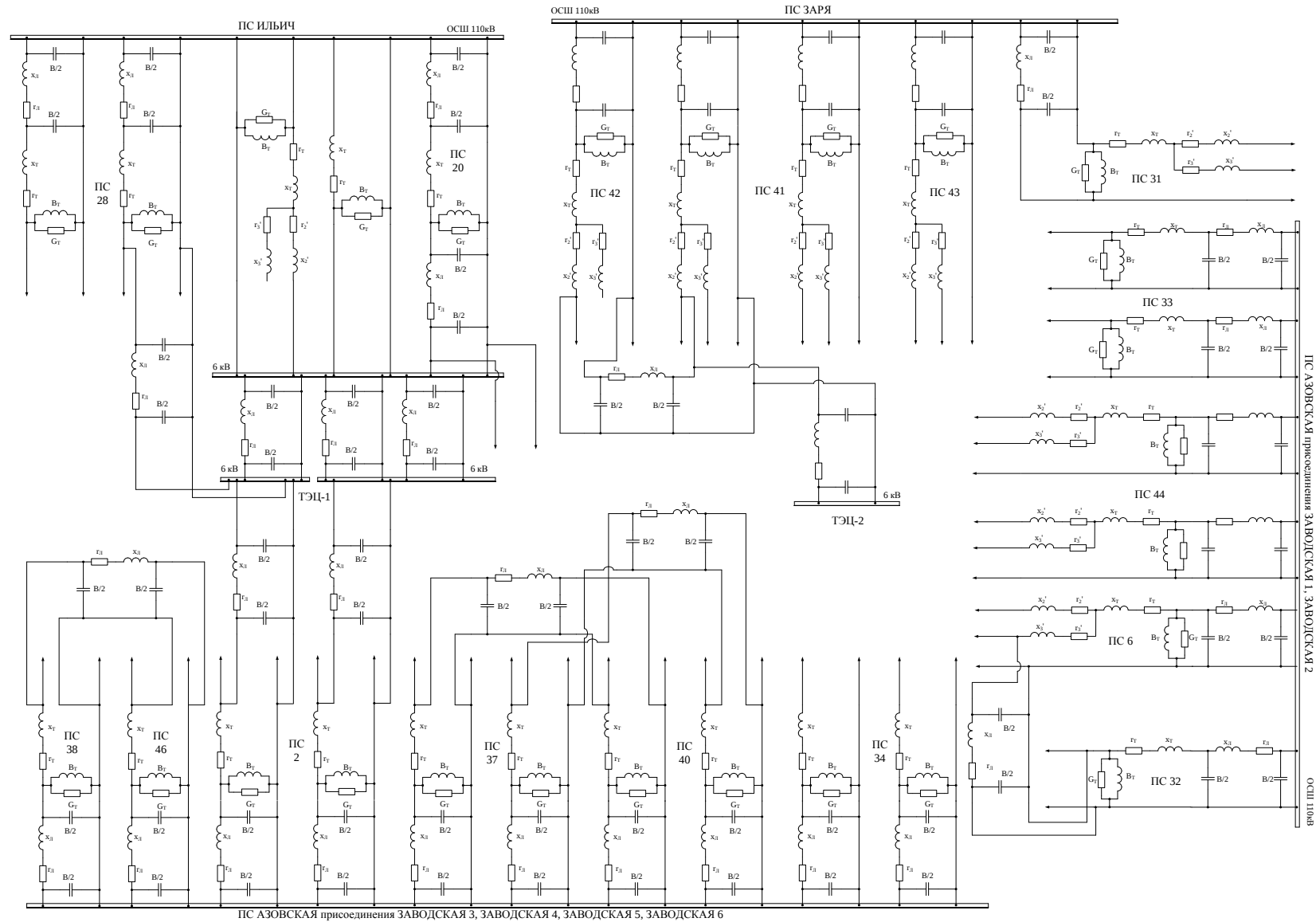


Рисунок 5.4 – Схема замещения электрической сети ОАО "ММК"

### 5.2 Расчет сети при различных нагрузках

На первом этапе (см. рис. 3.1) с помощью разработанного программного обеспечения были рассчитаны неоптимальные режимы работы сети ОАО "ММК" с компенсацией реактивной мощности при коэффициенте реактивной мощности  $tg\varphi_{\Sigma} = 0,14$ . При номинальных нагрузках потери активной мощности составили  $\Delta P_{неопт} = 88560$  кВт. При максимальных нагрузках -  $\Delta P_{неопт} = 93220$  кВт.

Таблица 5.18 - Оптимальный режим сети при номинальных нагрузках

| № п/ст | Приведенные напряжения к ступени 110 кВ | Генерируемые мощности |                 |
|--------|-----------------------------------------|-----------------------|-----------------|
|        |                                         | активная, кВА         | реактивная, кВА |
| 28     | 114,2                                   | -                     | 42351           |
| 2      | 113,8                                   | -                     | 38136           |
| 6      | 114,5                                   | -                     | 14233           |
| 32     | 113,7                                   | -                     | 9541            |
| 38     | 113,9                                   | -                     | 14919           |
| 46     | 114,1                                   | -                     | 5351            |
| 37     | 114,3                                   | -                     | 3780            |
| 40     | 113,8                                   | -                     | 10280           |
| 44     | 114,2                                   | -                     | 25306           |
| 34     | 114,5                                   | -                     | 15570           |
| 33     | 114,6                                   | -                     | 24905           |
| 20     | 113,8                                   | -                     | 45693           |
| 31     | 114,2                                   | -                     | 13560           |
| 43     | 114,1                                   | -                     | 10751           |
| 42     | 114,3                                   | -                     | 13341           |
| 41     | 114,2                                   | -                     | 65340           |
| ТЭЦ-1  | 114,8                                   | 15210                 | 6800            |
| ТЭЦ-2  | 115,1                                   | 32155                 | 12751           |

Проведя расчет оптимального режима сети ОАО "ММК" при номинальных нагрузках, было получено минимальное значение потерь активной мощности в сети  $\Delta P_{опт} = 84500$  кВт. Эффект разработанного

программного обеспечения оценивается по снижению потерь активной мощности на величину  $\Delta P' = \Delta P_{неопт} - \Delta P_{опт} = 88560 - 84500 = 4060$  кВт.

Таблица 5.19 – Оптимальный режим сети при максимальных нагрузках

| № п/ст | Приведенные напряжения к ступени 110 кВ | Генерируемые мощности |                 |
|--------|-----------------------------------------|-----------------------|-----------------|
|        |                                         | активная, кВА         | реактивная, кВА |
| 28     | 113,8                                   | -                     | 46283           |
| 2      | 113,3                                   | -                     | 40563           |
| 6      | 114,1                                   | -                     | 15621           |
| 32     | 113,1                                   | -                     | 10159           |
| 38     | 113,2                                   | -                     | 16105           |
| 46     | 113,8                                   | -                     | 6258            |
| 37     | 113,9                                   | -                     | 4106            |
| 40     | 113,5                                   | -                     | 12223           |
| 44     | 113,9                                   | -                     | 28412           |
| 34     | 114,0                                   | -                     | 16933           |
| 33     | 114,1                                   | -                     | 26122           |
| 20     | 113,4                                   | -                     | 47731           |
| 31     | 113,8                                   | -                     | 15303           |
| 43     | 113,7                                   | -                     | 116697          |
| 42     | 113,9                                   | -                     | 15105           |
| 41     | 113,9                                   | -                     | 68251           |
| ТЭЦ-1  | 114,7                                   | 16250                 | 7200            |
| ТЭЦ-2  | 115,0                                   | 34120                 | 14105           |

Проведя расчет оптимального режима сети ОАО "ММК" при максимальных нагрузках, было получено минимальное значение потерь активной мощности в сети  $\Delta P_{опт} = 89320$  кВт. Эффект разработанного программного обеспечения оценивается по снижению потерь активной мощности на величину  $\Delta P' = \Delta P_{неопт} - \Delta P_{опт} = 93220 - 89320 = 3900$  кВт.

## ***ВЫВОДЫ***

Разработанное программное обеспечение позволяет решить комплекс проблем, связанных с оптимальным распределением реактивной мощности по сети.

Разработанная адаптивная система управления позволяет осуществлять постоянный контроль над нагрузками подстанций и конфигурацией сети, оперативно реагировать на какие-либо изменения и производить точный расчет для нового режима.

Внедрение представленного комплекса, состоящего из программного обеспечения и адаптивной системы управления, на любое предприятие с любой сложно замкнутой сетью, в данном случае, на ОАО "ММК" позволяют:

- значительно уменьшить потери энергии;
- поддерживать постоянный уровень напряжения у приемного конца сети;
- значительно уменьшить денежные затраты на потребление и генерацию реактивной мощности.

Минимум аппаратного обеспечения, необходимого для информационного режима работы адаптивной системы управления, и совместимость системы с любым оборудованием, уже установленного на подстанциях, позволяет сделать вывод о целесообразности применения разработанного программного обеспечения на ОАО "ММК" и других крупных промышленных предприятий.

## ПЕРЕЧЕНЬ ССЫЛОК

1. В.Н. Костин.: Оптимизационные задачи электроэнергетики: Учеб. пособие. – СПб.: СЗТУ, 2003 – 120с.
2. В.Г. Карманов. Математическое программирование. – М.: Наука, 1980 – 256с.
3. Дж. Деннис, Р. Шнабель. Численные методы безусловной оптимизации и решения нелинейных уравнений. – М.: Мир, 1988 – 440с.
4. З.Н. Бененсон, М.Р. Елистратов, Л.К. Ильин и др. Моделирование и оптимизация на ЭВМ радиоэлектронных устройств. – М.: Радиосвязь, 1981 – 272с.
5. С. Гилл, У. Мюррей. Численные методы условной оптимизации. – М.: Мир, 1977 – 339с.
6. Васильев С.П. Численные методы решения экстремальных задач. – М.: Наука, 1980 – 518с.
7. И.Т. Черноруцкий. Методы оптимизации в теории управления, С.-П.: Питер, 2004 – 226с.
8. В.Е. Болконин, П.И. Чинаев. Анализ и синтез систем автоматического управления на ЭВМ. Алгоритмы и программы. – М.: Радио и связь, 1986 – 248с.
9. В.Н. Костин, Е.В. Распопов, Е.А. Родченко. Передача и распределение электроэнергии: Учеб. пособие. – СПб.: СЗТУ, 2003 – 147с.
10. Барзам А.Б. Системная автоматика. – М.: Энергоатомиздат, 1989. – 446с.
11. Баркан Я.Д., Орехов Л.А. Автоматизация энергосистем: учебное пособие для студентов вузов. – М.: Высш. школа, 1981. – 271с.
12. А. А. Черносивтов. Курс MCSD Visual C++ 6.0 и MFC, С.-П.: Питер, 2000 – 554с.
13. Веников В.А. Управление переходными режимами в электрических системах. – М.: Высшая школа, - 1972. – 352с.
14. [www.vdt-automation.com.ua](http://www.vdt-automation.com.ua)

## *Приложение А*

### *Программа расчета оптимальных режимов*

```

#include <stdio.h>
#include <complex.h>
#include <iostream.h>
#include <conio.h>
#include <math.h>

int luc ( int, complex[][31], int[], double );
void rluc ( int, complex[][31], int[], complex[] );
void gauss ( int n, complex[][31], complex[], complex * );
void rasis ( int, int, complex[], complex[], complex[], int[], int[] );
void start ( int, int, int, complex[][31], complex[],
             complex[], int[], int[], int[], int );
void vard ( int, int, int, complex[][31], complex[],
            complex[], int[], int[], int[] );
void varm ( int, int, int, complex[][31], complex[],
            complex[], int[], int[], int[] );
void prejc ( int, int, int, complex[], complex[] );
void rasis ( int, int, complex[], complex[], complex[], int[], int[] );
void pryzc ( int, complex[], int[], int[], int );
void foryzc ( int, int, complex[][31], complex[], int[], int[] );
double w;
void main ( void )
{
    complex a[30][31], b[30], yz[90], tok[90], s;
    int imax[90], imin[90], irow[30];
    int n, nyz, nej, istop, i, j;
    double f;
    do
    {
        clrscr ( );
        scanf ( "%i4 %i4 %i4 %i4", &f, &n, &nyz, &nej, &istop );
        w = 2 * M_PI * f;
        start ( n, nyz, nej, a, b, yz, imax, imin, irow, 0 );
        for ( i = 0; i < nyz; i++ )
        {
            yz[i] = 1.0 / yz[i];

```

```

    cout << yz[i];
    if ( ! ( i % 4 ) ) cout << '\n';
}
for ( i = 0; i < nyz; i++ )
    cin >> imax[i] >> imin[i];
cin >> n;
vard ( n, nyz, nej, a, b, yz, imax, imin, irow );
rais ( n, nyz, b, yz, tok, imax, imin );
}
while ( istop );
//for ( i = 0; i < n; i++ )
//for ( j = 0; j <= n; j++ )
//cin >> a[i][j];
//for ( i = 0; i < n; i++ )
//b[i] = a[i][n];
//for ( i = 0; i < n; i++ )
//{
//for ( j = 0; j <= n; j++ )
//cout << a[i][j];
//cout << '\n';
//}
//for ( i = 0; i < n; i++ )
//cout << b[i];
//luc ( n, a, irow, 0.00001 );
//rluc ( n, a, irow, b );
//cout << '\n';
//gauss ( n, a, b, &s );
//for ( i = 0; i < n; i++ )
//cout << b[i];
}

```

```

int luc ( int n, complex a[][31], int irow[], double eps )
{
    int i, il, j, k, l, n1;
    complex prom;

    for ( i = 0; i < n; i++ )

```

```

    irow[i] = i;
    n1 = n - 1;
    for ( i = 0; i < n1; i++ )
    {
        i1 = i + 1;
        l = i;
        for ( k = i1; k < n; k++ )
            if ( abs ( a[k][i] ) > abs ( a[l][i] ) ) l = k;
        if ( abs ( a[l][i] ) < eps ) return 0;
        if ( l > i )
        {
            for ( j = 0; j < n; j++ )
            {
                prom = a[i][j];
                a[i][j] = a[l][j];
                a[l][j] = prom;
            }
            irow[i] = l;
        }
        for ( j = i1; j < n; j++ )
            a[i][j] = a[i][j] / a[i][i];
        for ( k = i1; k < n; k++ )
            for ( j = i1; j < n; j++ )
                a[k][j] -= a[k][i] * a[i][j];
    }
    return 1;
}

```

```

void rluc ( int n, complex a[][31], int irow[], complex b[] )
{
    int i, j;
    complex sum;

    for ( i = 0; i < n; i++ )
        if ( i != irow[i] )
        {

```

```

    sum = b[i];
    b[i] = b[irow[i]];
    b[irow[i]] = sum;
}
for ( i = 0; i < n; i++ )
{
    sum = b[i];
    for ( j = 0; j < i; j++ )
        sum -= a[i][j] * b[j];
    b[i] = sum / a[i][i];
}
for ( i = n - 2; i >= 0; i-- )
{
    sum = b[i];
    for ( j = i+1; j < n; j++ )
        sum -= a[i][j] * b[j];
    b[i] = sum;
}
}

```

```

void gauss ( int n, complex a[][31], complex x[], complex *s )
{
    int i, j, k, l, k1, n1;
    complex r;
    n1 = n + 1;
    for ( k = 0; k < n; k++ )
    {
        k1 = k + 1;
        *s = a[k][k];
        j = k;
        for ( i = k1; i < n; i++ )
        {
            r = a[i][k];
            if ( abs ( r ) > abs ( *s ) )
            {
                *s = r;
                j = i;
            }
        }
    }
}

```

```

    }
}
if ( *s == complex ( 0.0, 0.0 ) ) break;
if ( j != k )
    for ( i = k; i < n1; i++ )
    {
        r = a[k][i];
        a[k][i] = a[j][i];
        a[j][i] = r;
    }
for ( j = k1; j < n1; j++ )
    a[k][j] /= *s;
for ( i = k1; i < n; i++ )
{
    r = a[i][k];
    for ( j = k1; j < n1; j++ )
        a[i][j] -= a[k][j] * r;
}
}
if ( *s != complex ( 0.0, 0.0 ) )
    for ( i = n - 1; i >= 0; i-- )
    {
        *s = a[i][n];
        for ( j = i + 1; j < n; j++ )
            *s -= a[i][j] * x[j];
        x[i] = *s;
    }
return;
}

```

```

void start ( int n, int nyz, int nej, complex a[][31], complex b[],
            complex yz[], int imax[], int imin[], int irow[], int iy )
{
    int i;

    pryzc ( nyz, yz, imax, imin, iy );
    for ( i = 0; i < nyz; i++ )

```

```

{
    cout << yz[i];
    if ( ! ( i % 4 ) ) cout << '\n';
}
vard ( n, nyz, nej, a, b, yz, imax, imin, irow );
}

```

```

void vard ( int n, int nyz, int nej, complex a[][31], complex b[],
            complex yz[], int imax[], int imin[], int irow[] )
{
    int i, j;

    foryzc ( n, nyz, a, yz, imax, imin );
    //for ( i = 0; i < n; i++ )
    //{
        //for ( j = 0; j < n; j++ )
            //cout << a[i][j];
        //cout << '\n';
    //}
    if ( luc ( n, a, irow, 1.0e-5 ) )
        varm ( n, nyz, nej, a, b, yz, imax, imin, irow );
    else
        printf ( "\nВ@3-@Г aГиГ-ЁГ -Г ъ@§-@|-@ " );
    return;
}

```

```

void varm ( int n, int nyz, int nej, complex a[][31], complex b[],
            complex yz[], int imax[], int imin[], int irow[] )
{
    int i;

    prejc ( n, nej, nyz, b, yz );
    for ( i = 0; i < n; i++ )
    {
        cout << b[i];
        if ( ! ( i % 4 ) ) cout << '\n';
    }
}

```

```

rluc ( n, a, irow, b );
for ( i = 0; i < n; i++ )
{
    cout << b[i];
    if ( ! ( i % 4 ) ) cout << '\n';
}
return;
}

```

```

void pryzc ( int nyz, complex yz[], int imax[], int imin[], int iy )
{
    int i, indrlc;
    complex rlc;

    for ( i = 0; i < nyz; i++ )
    {
        cin >> rlc >> indrlc >> imax[i] >> imin[i];
        yz[i] = rlc;
        if ( indrlc == 2 )
            yz[i] = complex ( 0.0, -1000000 / ( w * imag ( rlc ) ) );
        if ( indrlc == 1 )
            yz[i] = complex ( 0.0, w * imag ( rlc ) * 0.001 );
        if ( indrlc == 3 ) yz[i] = polar ( real ( rlc ), imag ( rlc ) );
        cout << "YZ[" << i+1 << "]= " << yz[i];
    }
}

```

```

void foryzc ( int n, int nyz, complex a[][31], complex yz[],
              int imax[], int imin[] )
{
    int i, j, k, kolstr;

    for ( i = 0; i < n; i++ )
    {
        for ( j = 0; j < n; j++ )
            a[i][j] = complex ( 0.0, 0.0 );
        //cout << '\n';
    }
}

```

```

    }
    for ( k = 0; k < nyz; k++ )
    {
        i = imax[k];
        j = imin[k];
        a[i-1][i-1] += yz[k];
        if ( j )
        {
            a[j-1][j-1] += yz[k];
            a[i-1][j-1] -= yz[k];
            a[j-1][i-1] = a[i-1][j-1];
        }
    }
    clrscr();
    cout << "YZ[i][j]=";
    kolstr = 0;
    for ( i = 0; i < n; i++ )
    {
        for ( j = 0; j < n; j++ )
        {
            cout << "\nYZ[" << i+1 << "][" << j+1 << "]= " << a[i][j];
            kolstr++;
            if ( kolstr == 22 )
            {
                kolstr = 0;
                getchar ( );
                clrscr();
                cout << "YZ[i][j]=";
            }
        }
    }
    return;
}

```

```

void prejc ( int n, int nej, int nyz, complex b[], complex yz[] )
{
    int i, ind1, ind2, imin, imax, iyz;

```

complex ej;

```

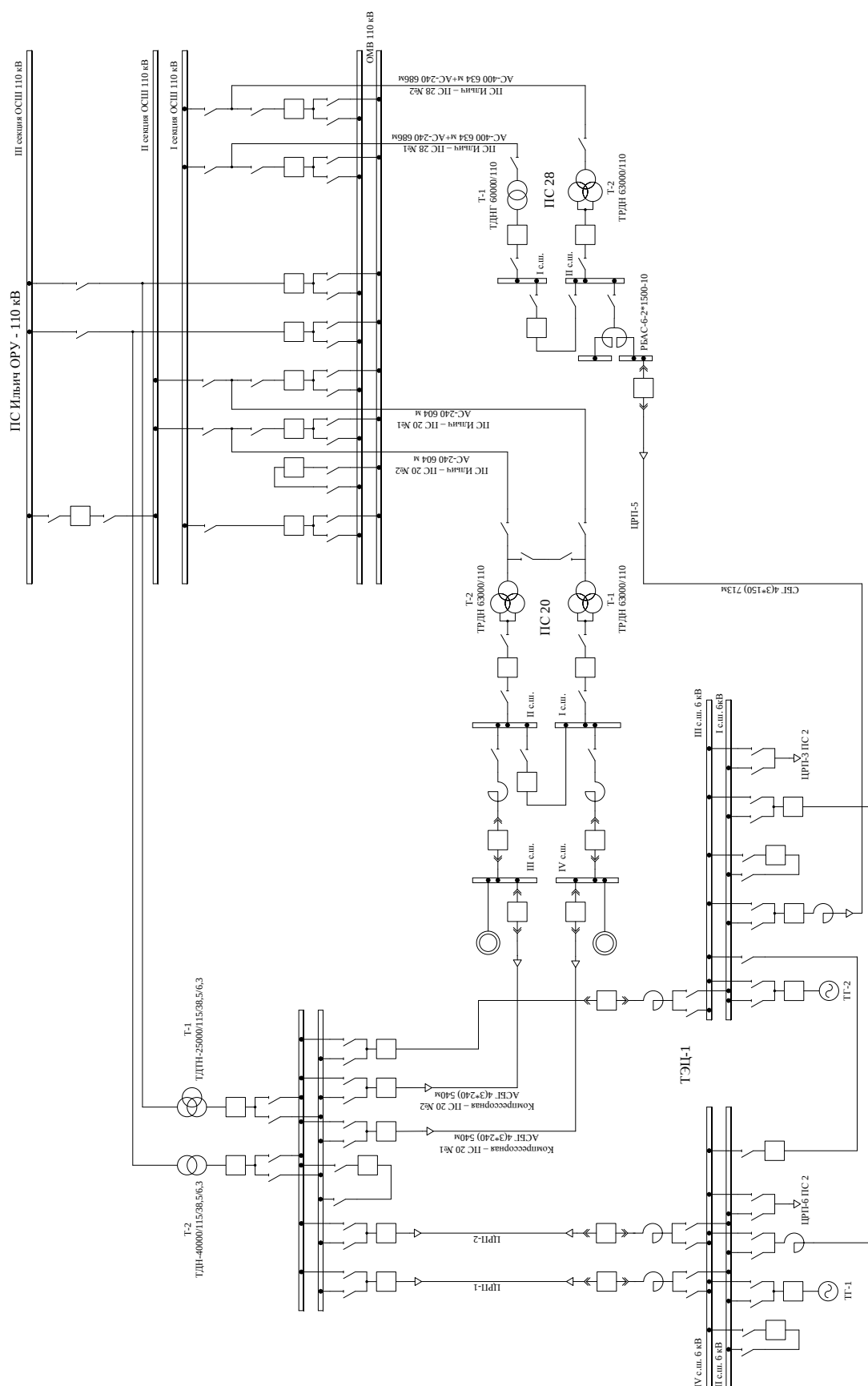
for ( i = 0; i < n; i++ )
    b[i] = complex ( 0.0, 0.0 );
for ( i = 0; i < nej; i++ )
{
    cin >> ej >> iyz >> imax >> imin >> ind1 >> ind2;
    if ( ind1 )
        ej = polar ( real ( ej ), imag ( ej ) );
    if ( ind2 ) ej *= yz[iyz-1];
    cout << "ei= " << ej;
    b[imax-1] += ej;
    if ( imin ) b[imin-1] -= ej;
}
return;
}

```

```

void rasis ( int n, int nyz, complex b[], complex yz[],
             complex tok[], int imax[], int imin[] )
{
    int i, j, k;
    for ( k = 0; k < nyz; k++ )
    {
        i = imax[k];
        j = imin[k];
        if ( j )
            tok[k] = ( b[i-1] - b[j-1] ) * yz[k];
        else
            tok[k] = b[i-1] * yz[k];
    }
    for ( k = 0; k < nyz; k++ )
        cout << "I[" << k+1 << "]= " << tok[k];
    return;
}

```







## Приложение В

### Структурная схема системы управления

