

# Методика выбора информационной структуры корпоративной компьютерной сети

А.В. Вишнеков, Е.М. Иванова, Ю.Л. Леохин

**Аннотация.** В статье рассматриваются вопросы выбора информационной структуры корпоративной компьютерной сети на основе применения методов поддержки принятия решений, приводится методика расчета параметров и критериев оценки информационной структуры при принятии проектного решения.

**Ключевые слова:** информационная структура, корпоративная компьютерная сеть, методы поддержки принятия решений, альтернативное проектное решение, критерии оценки.

## Введение

В настоящее время с учетом масштабности, высокой степени гетерогенности, использования глобальных связей проектирование корпоративных сетей является трудно формализуемым процессом. Этот факт объясняет отсутствие в настоящее время универсальных методик проектирования корпоративных сетей. В большинстве случаев сетевые интеграторы опираются на свой опыт при проектировании сети. Тем не менее, учитывая большой накопленный опыт проектирования компьютерных сетей, были определены следующие типовые этапы выполнения сетевых проектов [1-3].

1. *Анализ требований.* В результате анализа определяются основные цели предприятия, достижение которых позволяет повысить конкурентоспособность предприятия.

2. *Разработка бизнес-модели* предприятия. Бизнес-модель или функциональная модель производства описывает основные, административные и вспомогательные бизнес-процессы предприятия, информационные потоки между

подразделениями, иерархические взаимоотношения между подразделениями и представляет собой структурированное отображение функций производственной системы, среды, информации и объектов, связывающих эти функции.

3. *Разработка информационной структуры.* Информационная структура сети представляет собой совокупность информационных ресурсов корпоративной системы (источники и приемники информации), размещенные на узлах сети, и информационные потоки между узлами, возникающие при решении прикладных задач. Под термином узел информационной структуры понимается место размещения ресурсов (приложения, базы данных), где обеспечивается его работа. Располагая данными об информационной структуре сети, можно принимать решения об организации каналов связи между узлами сети, определять необходимые параметры каналов связи и сетевого оборудования, формировать техническую структуру сети.

4. *Разработка технической структуры.* Техническая структура корпоративной сети представляет собой совокупность технических

средств, необходимых для реализации проекта корпоративной сети. На данном этапе определяются технические параметры компонентов сети - полный функциональный набор необходимых аппаратных и программных средств без конкретизации марок и моделей оборудования.

5. Разработка *физической структуры*. Физическая структура корпоративной сети представляет собой подробное описание технических и программных средств, их количества, технических параметров и способов взаимодействия. Таким образом, физическая структура является конкретизацией технической структуры сети, в которой в соответствии с техническими параметрами, задаваемыми в технической структуре, выбраны конкретные сетевые устройства, протоколы и прочие сетевые технические средства.

6. *Моделирование и оптимизация*. На данном этапе производится моделирование с целью оценки характеристик функционирования корпоративной сети и их оптимизации.

7. *Установка и наладка*. Данный этап подразумевает координирование поставок от субподрядчиков, управление конфигурированием, установку и наладку оборудования, обучение персонала.

8. *Тестирование*. На этом этапе должны проводиться приемочные испытания, оговоренные в контракте с интегратором.

9. *Сопровождение и эксплуатация* корпоративной сети. Этот этап не имеет четко определенных временных границ, а представляет собой непрерывный процесс.

С учетом опыта конкретного администратора сети часть этапов может пропускаться при проектировании, часть объединяться. Например, бизнес-модель предприятия может быть реализована в виде информационной модели, учитывающей информационные потоки между подразделениями предприятия.

На этапах формирования *информационной, функциональной и технической структур* возможен синтез нескольких вариантов структур с отличными друг от друга параметрами (Рис. 1). Прежде чем переходить к следующему этапу проектирования необходимо выбрать наиболее рациональную структуру на текущем этапе, используя, например, методы теории принятия

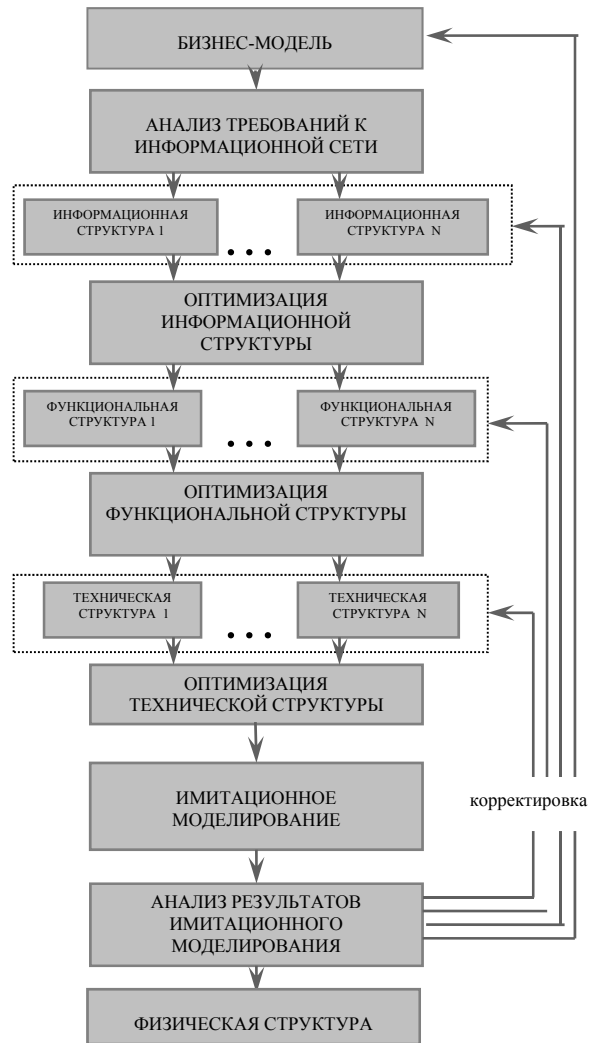


Рис. 1. Этапы проектирования корпоративной компьютерной сети

решения. Такой подход позволит повысить эффективность проектирования корпоративной компьютерной сети (ККС).

Выбор информационной/функциональной/технической структуры для проекта ККС представляет собой достаточно сложную многокритериальную задачу, причём критерии оценки структур могут носить взаимно противоречивый характер. Авторами предлагается методика принятия проектных решений, реализация которой рассматривается на примере разработки информационной структуры компьютерной сети. Данная задача может быть отнесена к классу задач принятия решений – задач оценки и сравнения многокритериальных альтернатив.

## 1. Методика принятия решения по выбору ИС ККС

Целью использования методов поддержки принятия решений на этом этапе является обеспечение процесса выбора наиболее рациональной информационной структуры.

Число возможных вариантов структур может быть значительным – несколько десятков. Поэтому перед разработчиком стоит задача выбора наиболее подходящего варианта по заданным критериям, для решения которой целесообразно использование методов поддержки принятия решений [4]. Применение методов поддержки принятия решений позволяет автоматизировать как процесс генерации возможных вариантов информационных структур, так и процесс принятия решений по выбору одной из них.

К числу наиболее часто используемых критериев для выбора рациональной структуры КМС можно отнести следующие [5-9]:

- K1. Отказоустойчивость
- K2. Время решения задач
- K3. Стоимость проекта
- K4. Загрузка узлов
- K5. Время задержки
- K6. Количество ошибок в канале
- K7. Время доставки
- K8. Время реакции сети
- K9. Вероятность доставки пакета узлу назначения без искажения
- K10. Вероятность потери пакетов

Рассмотрев особенности поставленной задачи принятия решения, можно сделать вывод, что число критериев отбора невелико ( $\leq 10$ ), а число альтернативных решений может быть достаточно большим ( $\geq 10$ ). Методику решения данной задачи предлагается представить в виде последовательности нескольких шагов.

Шаг 1. Расчёт весов критериев для ранжирования их по степени важности. Для решения данной задачи предлагается использовать процедуру расчёта весов, аналогичную процедуре метода аналитических иерархий [4]. Использование данной процедуры оправданно, так число критериев оценки информационных структур ККС, как указано выше, не превышает 10 и вероятность получения несогласованной матрицы попарного сравнения критериев по степени важности мала. Если число критериев велико

( $\geq 10$ ), то применение данной процедуры нецелесообразно.

Шаг 2. Сокращение множества рассматриваемых альтернатив. Для окончательного выбора необходимо обращение к суждениям экспертов, а значит, применение групповых методов поддержки принятия решений, т.к. информационная структура – это сложный технический объект. Для возможности использования групповых методов количество альтернатив не должно быть слишком велико (не более 5-7). Иначе возможно появление несогласованных оценок экспертов. Для сокращения множества рассматриваемых альтернатив предлагается использование метода лексикографического полуупорядочивания [10].

Шаг 3. Выбор конечной информационной структуры для проекта КМС на основе использования экспертных процедур. Для решения данной задачи целесообразно применения метода минимального расстояния [11]. Число альтернативных информационных структур ККС на данном шаге не может превышать 10 (что является ограничением на размерность матрицы сравнения альтернатив), так как в противном случае вычислительная сложность и время решения задачи резко возрастают.

Для сравнения вариантов информационных структур необходимо получить числовые или лингвистические оценки ИС ККС по названным выше критериям. Некоторые из оценок легко определяются разработчиками/руководителями, для других необходимо провести дополнительные расчёты. Например, для расчёта параметров потоков данных между узлами информационной структуры сети, среднего времени решения задач необходимо выполнить математическое описание синтезируемой информационной структуры. Математический аппарат для описания информационной структуры представлен ниже [12].

### 1.1. Описание и расчёт характеристик информационной структуры корпоративной сети

Источниками потоков данных в сети являются пользователи и приложения. Пользователи инициируют работу приложений, которые, в свою очередь, обращаются к базам данных, к другим приложениям, передавая и получая различную информацию.

Работа приложений определяется задачами, которые решаются системой. Число задач в системе обозначим  $L$ .

Будем считать, для упрощения, что каждое приложение, независимо от того, в какой задаче оно используется, работает всегда одинаково. Так что, если, в зависимости от задачи, меняется работа приложения, то будем считать, что это уже другое приложение.

Под приложением будем понимать программу, которая запускается при решении задачи, программа может быть как специальной, написанной для решения задачи, так и общесистемной, предназначенной для выполнения стандартных процедур, которые требуются при решении задачи.

В каждой системе имеются хранилища данных, которые в нашем случае будем называть базами данных; их число в системе обозначим  $R$ . Считаем, что базы данных сформированы, состав данных определен и определены приложения, которые работают с базами данных.

Пусть число узлов сети –  $M$ , количество пользователей системы –  $N$ . Пусть в системе функционирует  $D$  различных приложений. Приложения предназначены для решения прикладных задач, которые необходимы для обслуживания пользователей корпоративной системы.

Каждая задача номер  $k$  характеризуется следующим набором параметров:  $S_k = \{p_k, d_k, u_k, W_k\}$ ,  $k=1, 2, \dots, L$ . Вектор-строка  $p_k = (p_{k1}, p_{k2}, \dots, p_{kD})$  определяет приложения, которые исполняются при выполнении задачи номер  $k$ , при этом  $p_{ki}=1$ , если приложение номер  $i$  выполняется при решении задачи номер  $k$ , и  $p_{ki}=0$ , если приложение номер  $i$  не выполняется при решении задачи номер  $k$  ( $i=1, 2, \dots, D; k=1, 2, \dots, L$ ). Для всех элементов вектор-строки  $p_k$  должно выполняться условие:  $\sum_{i=1}^D p_{ki} \geq 1$ , означающее, что при решении задачи номер  $k$  используется хотя бы одно приложение. Из вектор-строк  $p_k = (p_{k1}, p_{k2}, \dots, p_{kD})$  можно составить матрицу  $P = \|p_{ki}\|$ , задающую связи между задачами системы и приложениями.

Вектор-строка  $d_k = (d_{k1}, d_{k2}, \dots, d_{kR})$  определяет базы данных, которые используются при выполнении задачи номер  $k$ , при этом  $d_{ki}=1$ , если база данных номер  $i$  используется при решении задачи номер  $k$ , и  $d_{ki}=0$ , если база данных

номер  $i$  не используется при решении задачи номер  $k$  ( $i = 1, 2, \dots, R; k = 1, 2, \dots, L$ ). Для всех элементов вектор-строки  $d_k$  должно выполняться условие:  $\sum_{i=1}^R d_{ki} \geq 0$ , означающее, что при решении задачи номер  $k$  может не использоваться ни одной базы данных. Из вектор-строк  $d_k = (d_{k1}, d_{k2}, \dots, d_{kR})$  можно составить матрицу  $D = \|d_{ki}\|$ , задающую связи между задачами системы и базами данных.

Вектор-строка  $u_k = (u_{k1}, u_{k2}, \dots, u_{kN})$  определяет множество пользователей системы, которым требуется запускать задачу номер  $k$ , при этом  $u_{kj}=1$ , если пользователь номер  $j$  запускает задачу номер  $k$ , и  $u_{kj}=0$ , если пользователь номер  $j$  не запускает задачу номер  $k$  ( $j = 1, 2, \dots, N; k = 1, 2, \dots, L$ ). Для всех элементов вектор-строки  $u_k$  должны выполняться следующие условия.

1.  $\forall k = 1, 2, \dots, L; \sum_{j=1}^N u_{kj} \geq 1$ , т.е. каждая задача запускается хотя бы одним пользователем.

2.  $\forall j = 1, 2, \dots, N; \sum_{k=1}^L u_{kj} \geq 1$ , т.е. каждый пользователь запускает хотя бы одну задачу.

Из вектор-строк  $u_k = (u_{k1}, u_{k2}, \dots, u_{kN})$  можно составить матрицу  $U = \|u_{kj}\|$ , определяющую потребности пользователей системы в запуске задач.

Матрица  $W = \|w_{kij}\|$ , ( $i = 1, 2, \dots, D; j = 1, 2, \dots, D$ ) определяет последовательность запуска приложений при решении задачи номер  $k$ . Так,  $w_{kij}=1$ , если приложение номер  $j$  при решении задачи  $k$  запускается после окончания работы приложения номер  $i$ , и  $w_{kij}=0$ , если приложение номер  $j$  при решении задачи  $k$  не запускается после окончания работы приложения номер  $i$ . Кроме того,  $w_{kii}=1$ , если приложение номер  $i$  запускается первым при решении задачи  $k$ . Если приложение  $j$  при решении задачи  $k$  запускается последним, то  $\sum_{m=1}^D w_{kjm} = 0$  и  $\sum_{m=1}^D w_{kmj} \geq 1$ .

Каждый  $i$  пользователь характеризуется интенсивностью потока запросов на запуск задач в системе  $\lambda_{ij} \geq 0$  ( $i = 1, 2, \dots, N; j = 1, 2, \dots, L$ ). Здесь  $\lambda_{ij}$  – интенсивность потока запросов от пользователя номер  $i$  на запуск задачи номер  $j$ . Множество интенсивностей потоков запросов от пользователей на запуск задач будем задавать матрицей  $\Lambda = \|\lambda_{ij}\|$ . Очевидно, что  $\lambda_{ij}=0$ , если  $u_{ji}=0$ , т.е. интенсивность потока запросов на запуск задачи  $j$  от пользователя  $i$  равна нулю, если этот пользователь не запускает задачу.

Каждое приложение номер  $m$ , используемое при решении задачи  $k$ , будем задавать набором  $A_{km} = \{v_{km}, b_{km}\}$ ,  $k=1, 2, \dots, L$ ;  $m=1, 2, \dots, D$ .

Здесь вектор-строка  $v_{km} = (v_{km1}, v_{km2}, \dots, v_{kmR})$  определяет объемы данных, которыми обменивается приложение  $m$  с базами данных за один сеанс решения задачи  $k$ . Так,  $v_{kmr} \geq 0$  – объем данных, которыми обменивается приложение  $m$  с базой данных  $r$ . Из вектор-строк  $v_{km} = (v_{km1}, v_{km2}, \dots, v_{kmR})$  можно составить матрицы  $V_k = \|v_{kmr}\|$  ( $m=1, 2, \dots, D$ ;  $r=1, 2, \dots, R$ ), задающие объемы передаваемых данных между приложениями и базами данных при решении задач.

Вектор-строка  $b_{km} = (b_{km1}, b_{km2}, \dots, b_{kmD})$  определяет объемы данных, которыми обменивается приложение номер  $m$  с другими приложениями при решении задачи номер  $k$ . Так,  $b_{kmj} \geq 0$  – объем данных, которыми обменивается приложение  $m$  с приложением  $j$ . Из вектор-строк  $b_{km} = (b_{km1}, b_{km2}, \dots, b_{kmD})$  можно составить матрицы  $B_k = \|b_{kmj}\|$ , ( $m=1, 2, \dots, D$ ;  $d=1, 2, \dots, D$ ), задающие объемы передаваемых данных между приложениями.

Таким образом, интенсивности и объемы потоков данных в сети корпоративной системы определяются интенсивностями запуска задач пользователями и запускаемыми при этом приложениями. Здесь условимся, что объем передаваемых по сети данных задается в установленных единицах, например, байтах.

Зададим размещение приложений по узлам сети с помощью матрицы  $G_k = \|g_{ij}\|$ , где  $g_{ij} = 1$ , если приложение номер  $i$  установлено на узле номер  $j$  и  $g_{ij} = 0$ , если приложение номер  $i$  не установлено на узле номер  $j$  (здесь  $i = 1, 2, \dots, D$ ;  $j = 1, 2, \dots, M$ ). Для элементов матрицы  $G$  должны выполняться следующие условия.

1. Для любого  $j = 1, 2, \dots, M$  справедливо неравенство:  $\sum_{i=1}^D g_{ij} \geq 1$ , т.е. на каждом узле должно быть установлено хотя бы одно приложение, но на узле может быть установлено и несколько приложений.

2. Для любого  $i = 1, 2, \dots, D$  справедливо неравенство:  $\sum_{j=1}^M g_{ij} = 1$ , т.е. каждое приложение должно быть установлено только на одном узле.

Из приведенных условий следует, что общее количество установленных на узлах приложе-

ний равно общему количеству приложений  $\sum_{i=1}^D \sum_{j=1}^M g_{ij} = D$ .

Подключение пользователей к узлам задается матрицей  $H = \|h_{ij}\|$ , где  $h_{ij} = 1$ , если пользователь номер  $i$  подключен к узлу номер  $j$  и  $h_{ij} = 0$ , если пользователь номер  $j$  не подключен к узлу номер  $i$  ( $i=1, 2, \dots, N$ ;  $j=1, 2, \dots, M$ ). Для элементов матрицы  $H$  выполняются следующие условия.

1.  $\sum_{i=1}^D \sum_{j=1}^M h_{ij} = M$ , т.е. все пользователи системы должны быть подключены к узлам сети системы.

2.  $\forall i = 1, 2, \dots, N$ ;  $\sum_{j=1}^M h_{ij} = 1$ , т.е. каждый пользователь должен быть подключен только к одному узлу (этот узел является рабочим местом пользователя).

3.  $\forall j = 1, 2, \dots, M$ ;  $0 \leq \sum_{i=1}^N h_{ij} \leq N$ , т.е. к любому узлу может быть либо подключено несколько пользователей, либо ни одного.

Распределение баз данных по узлам сети задается матрицей  $S = \|s_{rm}\|$ , где  $s_{rm} = 1$ , если база данных номер  $r$  размещена на узле номер  $m$ , и  $s_{rm} = 0$ , если база данных номер  $r$  не размещена на узле сети номер  $m$ , ( $r = 1, 2, \dots, R$ ;  $m = 1, 2, \dots, M$ ). Для элементов матрицы должны выполняться следующие условия.

1.  $\sum_{r=1}^R \sum_{m=1}^M s_{rm} = R$ , т.е. все базы данных системы должны быть размещены на узлах сети системы.

2.  $\forall r = 1, 2, \dots, R$ ;  $\sum_{m=1}^M s_{rm} \geq 1$ , т.е. каждая база данных может быть размещена более чем на одном узле сети, что на практике соответствует, например, репликации данных.

3.  $\forall m = 1, 2, \dots, M$ ;  $\sum_{r=1}^R s_{rm} \geq 0$ , т.е. на любом узле может быть размещено либо несколько баз данных, либо ни одной. (Здесь распределенную базу данных будем рассматривать как совокупность отдельных баз данных).

## 1.2. Расчет параметров потоков данных между узлами информационной структуры сети

Используя введенные средства описания (определения, задания) информационной структуры сети в виде набора данных, можно определить параметры потоков данных между узлами сети.

Как уже отмечалось выше, источниками потоков данных являются запросы пользователей на запуск задач и запускаемые для решения за-

дач приложения. Обозначим  $\Lambda = \|\lambda_{ij}\|$  ( $i = 1, 2, \dots, N; j = 1, 2, \dots, L$ ) – матрицу интенсивностей потоков запросов пользователей на запуск задач, где  $\lambda_{ij} \geq 0$  – интенсивность потока запросов от пользователя  $i$  на запуск задачи  $j$ .

Значения элементов матрицы  $\Lambda$  определяются спецификой работы пользователей корпоративной сети, и будем считать их известными.

Очевидно, что потоки запросов пользователей сначала поступают на те узлы сети, к которым прикреплены пользователи. Закрепление пользователей за узлами задается матрицей  $\mathbf{H}$ , определенной ранее.

Суммарная интенсивность потока запросов на запуск задачи  $k$  –  $\lambda_k$  вычисляется по формуле:

$$\lambda_k = \sum_{i=1}^N \lambda_{ik}, k = 1, 2, \dots, L. \quad (1)$$

Если ввести вектор-строку интенсивностей запуска задач в системе  $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_L)$ , то ее можно вычислить, используя (1), по формуле:

$$\lambda = \mathbf{e}_N \Lambda, \quad (2)$$

где  $\mathbf{e}_N$  – единичный вектор-строка размерности  $N$ .

Рассмотрим вектор-строку  $\gamma = (\gamma_1, \gamma_2, \dots, \gamma_D)$ :

$$\gamma = \lambda \mathbf{P}, \quad (3)$$

где элемент номер  $j$  вычисляется по формуле:  $\gamma_j = \sum_{k=1}^L \lambda_k p_{kj}$ , что означает суммарную интенсивность запуска приложения номер  $j$  всеми задачами, решаемыми системой. Таким образом, вектор-строка  $\gamma$  определяет интенсивности запуска приложений в корпоративной сети.

Определим суммарные объемы данных, передаваемых между узлами сети при решении задачи  $k$ . Пусть  $\mathbf{Z} = \|z_{kij}\|$  – матрица, каждый элемент которой  $z_{kij}$  равен суммарному объему данных, передаваемому между узлами сети  $i$  и  $j$  при решении задачи  $k$ . Тогда можно записать

$$z_{kij} = \left[ \sum_{r=1}^D g_{ri} p_{kr} \left( \sum_{m=1}^D g_{mi} p_{km} b_{rm} \right) \right] + \left[ \sum_{r=1}^D g_{ri} p_{kr} \left( \sum_{m=1}^R s_{mi} d_{km} v_{rm} \right) \right], \quad (4)$$

где  $k=1, 2, \dots, L; i=1, 2, \dots, M; j=1, 2, \dots, M$ .

В формуле (4) первое слагаемое – объемы данных, передаваемые между всеми приложениями, используемыми задачей  $k$  и установленными на узлах  $i$  и  $j$ . Второе слагаемое – объемы

данных, передаваемые между всеми приложениями и базами данных, используемыми задачей  $k$  и установленными на узлах  $i$  и  $j$ .

Используя (4), можно определить интенсивности потоков данных между узлами сети при решении задачи  $k$ . Естественно, что эти интенсивности определяются интенсивностью запуска задачи  $k$  всеми пользователями сети.

Здесь считаем, что каждое приложение, используемое при решении задачи, запускается один раз, хотя это ограничение легко обойти.

Матрица интенсивностей потоков данных между узлами сети при решении задачи  $k$  вычисляются по формуле:

$$\mathbf{A}_k = \lambda_k \mathbf{Z}_k (k=1, 2, \dots, L). \quad (5)$$

Формулу (5) можно записать и так:  $\mathbf{A} = \|a_{kij}\|$  – где  $a_{kij} = \lambda_k z_{kij}$  ( $i=1, 2, \dots, M; j=1, 2, \dots, M$ ).  $a_{kij}$  – суммарная интенсивность потоков данных между узлами  $i$  и  $j$  при решении задачи  $k$ .

Теперь, используя (4) можно определить суммарные интенсивности потоков данных, передаваемых между информационными узлами. Эти интенсивности задаются матрицей:

$$\mathbf{A} = \sum_{k=1}^L \mathbf{A}_k \quad (5)$$

или  $\mathbf{A} = \|a_{ij}\|$  – где  $a_{ij} = \sum_{k=1}^L a_{kij} = \sum_{k=1}^L \lambda_k z_{kij}$  ( $i=1, 2, \dots, M; j=1, 2, \dots, M$ ) – суммарная интенсивность потоков данных, передаваемых от узла  $i$  к узлу  $j$ . Формулы (5) и (6) определяют загрузку каналов связи и коммуникационного оборудования сети.

Определим теперь нагрузку на информационные узлы. Будем исходить из того, что нагрузка на узел, на котором установлены приложения (базы данных) определяется интенсивностью потока запросов к этим приложениям (базам данных). Из формулы (3) известны интенсивности запуска приложений, установленных на узлах сети  $\gamma_j$  ( $j=1, 2, \dots, D$ ). Используя эти данные и данные о распределении приложений по узлам (матрица  $\mathbf{G}$ ), получим формулу для вычисления интенсивности потока запросов на запуск приложения  $j$ , установленного на узле  $i$ :

$$\beta_{ji} = \gamma_j g_{ji} (i=1, 2, \dots, D; j=1, 2, \dots, D). \quad (7)$$

Это равенство можно для всех  $j$  и  $i$  записать в матричной форме в виде:  $\mathbf{B}^* = \|\beta_{ji}\| = \Gamma_{dg} \mathbf{G}$ , где

$\Gamma_{dg} = \|\gamma^*_{ij}\|$  – диагональная матрица, у которой  $\gamma^*_{ij} = \gamma_i$  и  $\gamma^*_{ij} = 0$ , если  $i \neq j$ , ( $i=1,2,\dots,D; j=1,2,\dots,D$ ).

Очевидно, что если  $\beta_{ji} = 0$ , то это значит, что приложение номер  $j$  не установлено на узле номер  $i$ . Формула (7) позволяет вычислить значение интенсивности потока запросов на запуск приложения от всех задач.

Для вычисления интенсивности потока запросов на запуск приложения номер  $j$  на узле  $i$  только от задачи  $k$ , можно воспользоваться формулой:

$$\beta_{ki} = \lambda_k \cdot g_{ji} \cdot p_{kj} \quad (k=1,2,\dots,L; j=1,2,\dots,D; i=1,2,\dots,M). \quad (8)$$

Из (8) и (7) следует, что  $\beta_{ij} = \sum_{k=1}^L \beta_{kji}$ . Если на узле  $i$  установлена база данных, можно определить интенсивность потока запросов к базе данных  $j$  при решении задачи  $k$ :

$$\varphi_{kji} = \lambda_k \cdot s_{ji} \cdot d_{kj} \quad (k=1,2,\dots,L; j=1,2,\dots,R; i=1,2,\dots,M). \quad (9)$$

Из (9) получим формулу для вычисления суммарной интенсивности потока запросов к базе данных  $j$ , установленной на узле  $i$  при решении задач:

$$\varphi_{ji} = \sum_{k=1}^L \lambda_k s_{ji} d_{kj} = \sum_{k=1}^L \varphi_{kij} \quad (j=1,2,\dots,R; i=1,2,\dots,M) \quad (10)$$

Здесь также, если  $\varphi_{ji} = 0$ , то на узле  $i$  не установлена база данных номер  $j$ .

Из величин  $\varphi_{ji}$ , вычисляемых по формуле (10), можно составить матрицу  $\Phi = \|\varphi_{ji}\|$  ( $j=1,2,\dots,R; i=1,2,\dots,M$ ). Отметим, что формулы (7) и (10) предусматривают также случаи, когда на одном узле установлено несколько приложений или несколько баз данных. Тогда суммарная интенсивность потока запросов на запуск приложений, установленных на узле  $i$ , вычисляется по формуле:  $\beta_i = \sum_{j=1}^D \beta_{ji}$ , а суммарная интенсивность потока запросов к базам данных, установленным на узле  $i$ , вычисляется по формуле:  $\varphi_i = \sum_{j=1}^R \varphi_{ji}$ .

Приведем способ расчёта времени, затрачиваемого для решения каждой из задач и общего времени решения задач в сети. Расчёты ведутся при условии, что производительность каждого информационного узла одинаковая, и время решения задач в этом случае напрямую зависит от размещения приложений и баз данных по

информационным узлам. Чем больше приложений находится на одном узле, тем больше общее время решения задачи:

$$W_1 = \frac{\lambda_{sum} \cdot B_{2i}}{2 \cdot (1 - \lambda_{sum} \cdot B_{1i})}, \quad (11)$$

$$\lambda_{sum} = \lambda_i + \lambda_j; q_i = \frac{\lambda_i}{\lambda_{sum}}; q_j = \frac{\lambda_j}{\lambda_{sum}}; \quad (12)$$

$$W_2 = \frac{\lambda_{sum} \cdot (q_i \cdot B_{2i} + q_j \cdot B_{2j})}{2 \cdot (1 - (\lambda_{sum} \cdot (q_i \cdot B_{1i} + q_j \cdot B_{1j})))}. \quad (13)$$

Общее время решения задач рассчитывается по формуле:

$$W_{sum} = W_1 + B_{1i}; W_{sum1} = \sum_{i=1}^k W_{sumi}, \quad (14)$$

где:  $i, j$  – номера приложений;

$k$  – количество приложений и баз данных в одной задаче;

$W_1$  – время ожидания в очереди (для случая нахождения в информационном узле одного приложения или запроса к одной базе данных);

$W_2$  – время ожидания в очереди (для случая нахождения в информационном узле двух приложений или запросов к двум базам данных);

$W_{sum}$  – суммарное время обработки одного приложения или обработки запроса к одной базе данных;

$W_{sum1}$  – суммарное время обработки одной задачи;

$\lambda_i$  – интенсивность запроса  $i$ -ого приложения к  $i$ -ой базе данных;

$\lambda_j$  – интенсивность запроса  $j$ -ого приложения к  $j$ -ой базе данных;

$\lambda_{sum}$  – суммарная интенсивность запросов;

$B_{1i}$  – среднее время исполнения  $i$ -ого приложения или среднее время обслуживания запроса к  $i$ -ой базе данных;

$B_{2i}$  – второй момент времени исполнения  $i$ -ого приложения или второй момент времени обслуживания запроса к  $i$ -ой базе данных;

$B_{1j}$  – среднее время исполнения  $j$ -ого приложения или среднее время обслуживания запроса к  $j$ -ой базе данных;

$B_{2j}$  – второй момент времени исполнения  $j$ -ого приложения или второй момент времени обслуживания запроса к  $j$ -ой базе данных.

Используя приведенные выше формулы, можно рассчитать загрузки каналов и сетевых узлов и общие времена решения задач для трех вариантов информационных структур корпоративной компьютерной сети с целью выбора рационального варианта.

## 2. Примеры решения задачи

Предлагается рассмотреть пример выбора ИС ККС, для оценки которых отобраны следующие критерии:

- К1. Наилучшая отказоустойчивость;
- К2. Наименьшее время решения задач;
- К3. Минимальная стоимость проекта;
- К4. Минимальная загрузка узлов.

Рассмотрим на шаге 1 пример применения процедуры попарного сравнения критериев (согласно процедурам метода аналитических иерархий). Процедура может проводиться одним человеком – ЛПР (лицом, принимающим решение) и заключается в выставлении сравнительных оценок каждой паре критериев  $K_i - K_j$  по следующему правилу.

Оценка 1 выставляется обоим критериям, если они имеют равную важность для рассматриваемого проекта. Оценка 3 выставляется критерию  $K_i$ , если его важность для рассматриваемого проекта незначительно превосходит важность критерия  $K_j$  (тогда критерию  $K_j$  выставляется оценка – 1/3). При существенном превосходстве критерию  $K_i$  выставляется оценка 5 (критерию  $K_j - 1/5$  соответственно). При значительном превосходстве критерию  $K_i$  выставляется оценка 7 (критерию  $K_j - 1/7$  соответственно). При очень большом превосходстве критерию  $K_i$  выставляется оценка 9 (критерию  $K_j - 1/9$  соответственно).

Допустим, была получена следующая матрица попарного сравнения критериев (Рис. 2).

Вес  $W_i$  каждого  $i$ -го критерия в дальнейшем рассчитывается по формуле (15), где  $n$  – общее число критериев;  $i=j=1, \dots, n$ ;  $x_{ij}$  – оценка-элемент матрицы в строке  $i$  и столбце  $j$ .

$$X = \begin{pmatrix} 1 & 5 & 7 & 3 \\ 1/5 & 1 & 7 & 1 \\ 1/7 & 1/7 & 1 & 1/3 \\ 1/3 & 1 & 3 & 1 \end{pmatrix}$$

Рис.2. Матрица попарного сравнения критериев

$$W_i = \frac{\sqrt[n]{\prod_j x_{ij}}}{\sum_i \sqrt[n]{\prod_j x_{ij}}} \tag{15}$$

В результате расчетов были получены следующие веса критериев:  $W_1=0,57$ ;  $W_2=0,20$ ;  $W_3=0,05$ ;  $W_4=0,18$ .

Далее переходим к шагу 2 – сокращения числа альтернатив. Метод лексикографического полуупорядочивания подразумевает определение удовлетворительных интервалов на шкалах значений критериев и последовательное исключение альтернатив, чьи оценки по каждому критерию выходят за границы определенного интервала. Допустим, по заданным критериям определены следующие интервалы шкал значений (Табл. 1). Интервалы не указываются в конкретных значениях, а берутся в условных единицах, поскольку сильно зависят от условий выполнения конкретного проекта.

Применение метода полуупорядочивания начинается с наиболее важного критерия (с максимальным весом) – критерия К1 (вес  $W_1=0,57$ ), и сначала отбрасываются альтернативы с неудовлетворительными оценками по этому критерию ( $q < q_{\min}$  или  $q > q_{\max}$ ). Затем аналогичная процедура сокращения альтернатив проводится последовательно по менее значимым критериям: К2, К3, К4.

Табл. 1. Значения удовлетворительных интервалов на шкалах критериев

№ п.п.	Название критерия	Интервал удовлетворительных значений
К1	Отказоустойчивость (вероятность безотказной работы)	$q_{\min} \div q_{\max}$ (определяется ТЗ и проверяется на этапе моделирования; например, 0,99÷1,0)
К2	Время решения задач	$t_{\min} \div t_{\max}$ (например, 5÷15 условных единиц времени моделирования, см. расчеты выше)
К3	Стоимость проекта	$P_{\min} \div P_{\max}$ (например, не более 2 млн.)
К4	Загрузка узлов	$R_{\max}$ (например, не менее 80%)



Табл. 2. Оценки выбранных альтернатив по критериям отбора

Альтернатива	Значения по критериям			
	К1	К2	К3	К4
ИС1	$q_1$	$t_1$	$P_1$	$R_1$
ИС2	$q_2$	$t_2$	$P_2$	$R_2$
ИС3	$q_3$	$t_3$	$P_3$	$R_3$

Допустим, что после выполнения шага 2 остались следующие альтернативы (Табл. 2).

Используя приведенные выше формулы (1-14), можно провести расчеты параметров ИС ККС: общее время решения задач, отказоустойчивость, нагрузка узлов. Расчет стоимости проекта производится экономистами предприятия. Рассмотрим три примера альтернативных ИС ККС [13].

**Вариант сегмента ИС1.** Исходные данные для формирования описания информационной структуры: количество пользователей системы  $N=6$  (П1÷П6), количество информационных узлов  $M=10$  (ИУ1÷ИУ10), количество приложений  $D=7$  (Пр1÷Пр7), количество баз данных  $R=3$  (БД1÷БД3). При этом одна БД3 используется двумя задачами. Пользователи 3 и 6 задачи не запускают (Рис. 3).

На этой схеме более равномерно распределены потоки данных и в обмене данными участвуют все узлы. Для данной структуры справедливы будут следующие оценки. Для каждого приложения выделен отдельный информационный узел. Высокая отказоустойчи-

вость вследствие того, что выход из строя одного узла приведет к сбою в работе только одного приложения, длительное время решения задач (Табл. 4), минимальная нагрузка узлов из-за равномерного распределения потоков данных по узлам и высокая стоимость проекта, поскольку в работе задействованы все узлы.

**Вариант сегмента ИС2.** Исходные данные для формирования описания информационной структуры аналогичные:  $N=6$ ,  $M=10$ ,  $D=7$ ,  $R=3$ . При этом одна БД3 используется двумя задачами. Пользователи 3 и 6 задачи не запускают (Рис. 4).

Для данной структуры справедливы будут следующие оценки. Относительно низкая отказоустойчивость вследствие того, что на трёх информационных узлах работают по 2 приложения и выход из строя одного из этих узлов приводит к сбою двух приложений, среднее время решения задач (Табл. 4), максимальная нагрузка узлов из-за неравномерности распределения трафика, низкая стоимость проекта, поскольку для реализации двух узлов можно не приобретать программно-аппаратные средства.

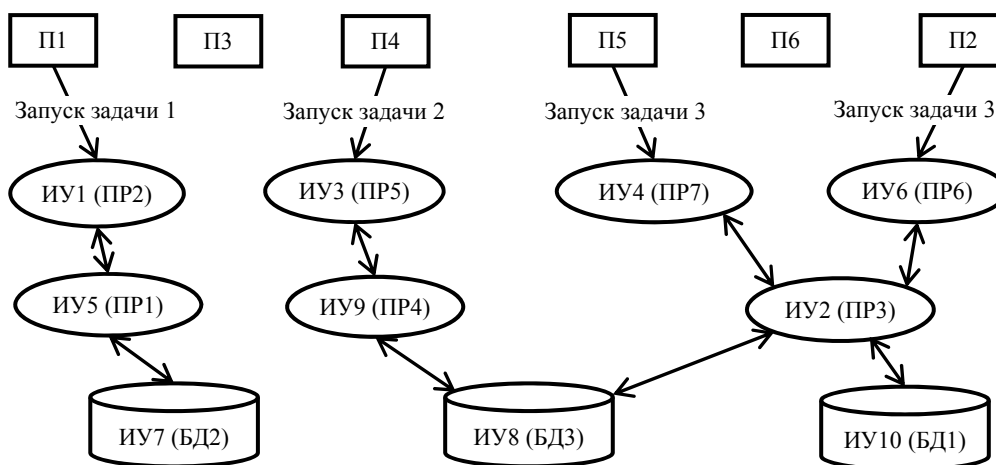


Рис. 3. Вариант сегмента информационной структуры ИС1

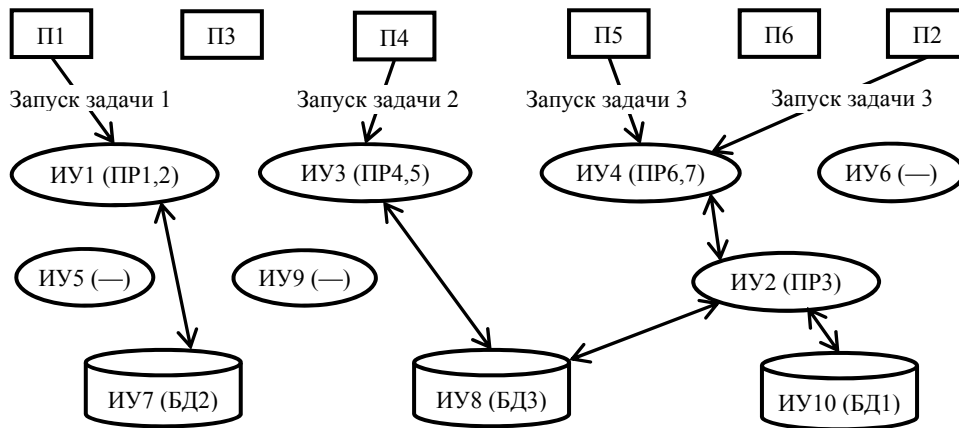


Рис. 4. Вариант сегмента информационной структуры ИС2

**Вариант сегмента ИС3.** Исходные данные для формирования описания информационной структуры аналогичные:  $N=6, M=10, D=7, R=3$ . При этом базы данных номер 1 и 3 располагаются на одном узле и используется двумя задачами (Рис. 5).

Для данной структуры будут справедливы следующие оценки. Низкая отказоустойчивость вследствие того, что две БД установлены на один информационном узле и его выход из строя приведет к сбою двух БД (что несколько лучше, чем в примере ИС2, но хуже, чем в примере ИС1), наименьшее время решения задач (Табл. 4), средняя загрузка узлов из-за неравномерности потоков (обращение к ИУ БД будет достаточно высока) и средняя стоимость проекта, поскольку незадействованным остается 1 узел.

Предположим, что среднее время выполнения приложений в узлах известно, также задано

среднее время выполнения запросов к базам данных (Табл. 3).

Согласно указанным исходным данным и формулам (11-14) получим следующие показатели времени решения каждой задачи и общего времени решения задач для заданных ИС1, ИС2, ИС3 (Табл. 4).

Анализируя исходные данные, можно прийти к следующим выводам. Анализ по критерию К1 (наилучшая отказоустойчивость) показал, что  $q_1 > q_3 > q_2$ , ранжировка альтернативных ИС будет следующей: ИС1, ИС3, ИС2 и наиболее предпочтительной является вариант №1.

Анализ трёх информационных структур по критерию К2 (наименьшее время решения задач)  $t_3 < t_2 < t_1$ , показал, что ранжировка альтернативных ИС будет следующей: ИС3, ИС2, ИС1 и вариант №3 считается наиболее предпочтительным, данная информационная структура является оптимальной.

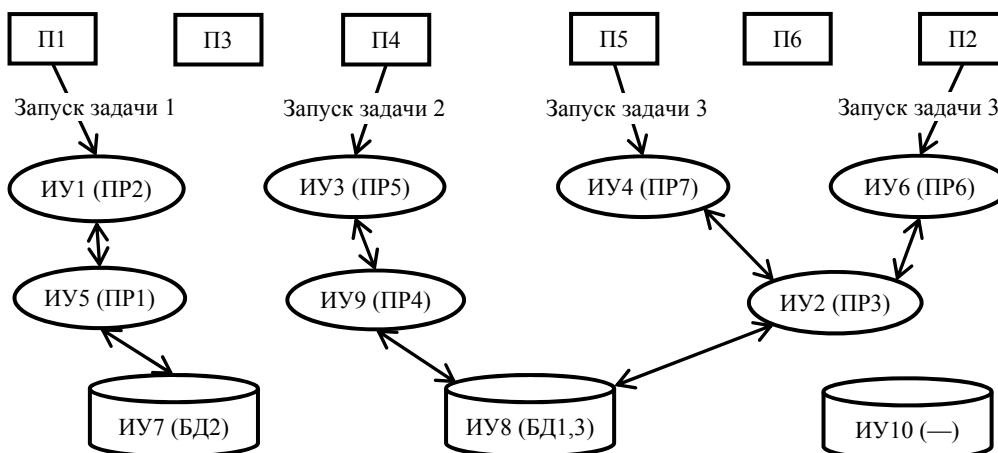


Рис.5. Вариант сегмента информационной структуры ИС3

Табл. 3. Среднее время выполнения приложений и запросов к базам данных в узлах

	Пр1	Пр2	Пр3	Пр4	Пр5	Пр6	Пр7	БД1	БД2	БД3
$B_{1i}$	1	1,5	1	2	1,3	1,7	1,4	1,2	2,4	2,3
$B_{2i}$	0,1	0,005	0,2	0,1	0,3	0,4	0,08	0,6	0,8	0,9

Табл.4. Результаты расчета 3-х вариантов информационных структур (ИС)

Вариант ИС <sub>i</sub>	Итоговое время решения			Общее время решения задач $t_i$
	задачи 1	задачи 2	задачи 3	
$i=1$	5.027	5.665	8.604	$t_1=19.296$
$i=2$	3.59	4.462	7.427	$t_2=15.479$
$i=3$	5.027	3.662	6.125	$t_3=14.814$

Анализ по критерию К3 (минимальная нагрузка узлов) показал, что  $R_1 > R_3 > R_2$ , ранжировка альтернативных ИС будет следующей: ИС1, ИС3, ИС2 и наиболее предпочтительной является вариант №1.

Допустим также, что по критерию К4 были получены следующие результаты оценки:  $P_2 < P_3 < P_1$ , ранжировка альтернативных ИС будет следующей: ИС2, ИС3, ИС1 и наиболее предпочтительной является вариант №2.

Далее эксперты оценивают альтернативные структуры по своему усмотрению, комплексно оценивая значимость той или иной альтернативы по всей совокупности критериев, и ранжируют их по степени предпочтительности. Допустим, в оценке участвуют 4 эксперта ( $m=4$ ). В конечном счете, эксперты должны сформировать оценочные матрицы  $Y^1 \div Y^4$  попарного сравнения альтернатив (Рис. 3). В матрице каждый элемент  $y_{ij}$  показывает взаимную важность информационной структуры ИС<sub>i</sub> по отношению к ИС<sub>j</sub> по следующему правилу: лучше (1), хуже (-1), равнозначна (0). Пусть, например, были получены оценочные матрицы (Рис. 6).

Для каждой матрицы  $Y^1 \div Y^4$  рассчитывается ранжировка альтернатив по правилу: все элементы по строкам матриц складываются, и в ранжировке наилучшей считается альтернатива

с максимальной суммой, а наихудшей – с минимальной, равнозначными – с одинаковыми суммами. На основании этих матриц можно сделать вывод о том, как каждый из экспертов проранжировал по важности имеющиеся структуры.

- Ранжировка эксперта1 (по матрице  $Y^1$ ) = ИС3, ИС1, ИС2;
- Ранжировка эксперта2 (по матрице  $Y^2$ ) = [ИС1, ИС2, ИС3] (альтернативы 1, 2 и 3 равнозначны);
- Ранжировка эксперта3 (по матрице  $Y^3$ ) = [ИС1, ИС3], ИС2 (альтернативы 1 и 3 равнозначны);
- Ранжировка эксперта4 (по матрице  $Y^4$ ) = ИС2, ИС3, ИС1.

Применяя метод минимального расстояния, нужно найти компромиссное решение (итоговую ранжировку альтернатив по степени предпочтительности) с точки зрения удовлетворения мнений всех экспертов в равной степени [11]. Итоговая ранжировка соответствует итоговой матрице оценок  $Y^*$ , которая вычисляется по следующему правилу: величина суммы модулей разностей её элементов и элементов любой из матриц  $Y^1 \div Y^4$  должна быть минимальной. Если таких равноудаленных матриц получено несколько, то ЛПП вправе выбрать любую из

$$Y^1 = \begin{pmatrix} 0 & 1 & -1 \\ -1 & 0 & -1 \\ 1 & 1 & 0 \end{pmatrix}; \quad Y^2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad Y^3 = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}; \quad Y^4 = \begin{pmatrix} 0 & -1 & -1 \\ 1 & 0 & 1 \\ 1 & -1 & 0 \end{pmatrix}$$

Рис. 6. Матрицы попарного сравнения альтернатив

них или может применить метод поддержки принятия индивидуальных решений, например, метод аналитических иерархий.

Вычисление итоговой равноудаленной матрицы происходит методом перебора всех существующих матриц с нулевой главной диагональю и кососимметричных относительно неё же (значение «1» выше диагонали симметрично значению «-1» ниже диагонали, и наоборот). Такой перебор удобнее всего выполнить программно. Согласно исходным данным программные расчёты дают следующую ранжировку (от наиболее предпочтительной альтернативы к наименее): ИС1, ИС3, ИС2. Значит, наиболее предпочтительной является альтернативы ИС1.

## Заключение

В статье предложены математический аппарат, позволяющий описывать и рассчитывать параметры информационных структур ККС, и методика расчета критериев оценки информационной структуры. Рассмотрена методика выбора информационной структуры корпоративной компьютерной сети. Данная методика основана на последовательном выполнении процедур генерации альтернативных структур, процедур оценки и сравнения структур на основе расчета критериев оценки альтернативных вариантов и выбора итоговой информационной структуры в условиях многокритериальности на основе применения методов поддержки принятия решений. Использование данной методики позволяет провести обоснованный выбор наиболее рационального проектного решения с учетом мнений как ЛПР (руководителя проекта), так и экспертов (инженеров разработчиков, административных работников, специалистов по оценке стоимостных параметров проекта и др.).

**Вишнеков Андрей Владленович.** Академический руководитель образовательных программ магистратуры НИУ ВШЭ по направлению "Информатика и вычислительная техника". Окончил Московский институт электронного машиностроения в 1982 году. Профессор, доктор технических наук. Количество печатных работ: более 100. Область научных интересов: классификация СППР, средства САПР, применение вычислительной техники и других средств автоматизации проектирования. E-mail: avishnekov@hse.ru

**Иванова Елена Михайловна.** Доцент НИУ ВШЭ. Окончила Московский институт электронного машиностроения в 1992 году. Кандидат технических наук. Количество печатных работ: более 50. Область научных интересов: параллелизм вычислений, применение математических методов исследования в других областях знания в технических науках, классификация СППР, преподавание информатики. E-mail: emivanova@hse.ru

**Леохин Юрий Львович.** Профессор НИУ ВШЭ. Окончил Московский институт электронного машиностроения в 1982 году. Доктор технических наук. Количество печатных работ: более 100. Область научных интересов: компьютерные сети, системы управления. E-mail: yleokhin@hse.ru

Предлагаемая методика позволяет автоматизировать процесс принятия проектного решения, повысить обоснованность выбора и сократить общее время проектирования.

## Литература

1. Леохин Ю.Л., Бекасов В.Ю. Корпоративные сети: состояние, перспективы и тенденции. М.: Фонд «Качество», 2008. 123 с.
2. Олифер В.Г., Олифер Н.А. и др. Стратегическое планирование сетей масштаба предприятия. М: Центр Информационных Технологий, 2000. 680 с.
3. Олифер В.Г., Олифер Н.А. Компьютерные сети: принципы, технологии, протоколы. 5-е изд. СПб: Издательство «Питер», 2016. 992 с.
4. Ларичев О.И. Теория и методы принятия решений, а также Хроника событий в Волшебных странах: Учебник. Изд. 3-е, перераб. и доп. - М.: Логос, 2008, – 392 с.
5. Дж. Курроуз, К. Росс. Компьютерные сети. Нисходящий подход - М: Э, 2016. – 908 с.
6. Э. Таненбаум, Д. Уэзеролл. Компьютерные сети. – СПб: Питер, 2016. -955 с.
7. Уилсон Э. Мониторинг и анализ сетей. – М: ЛОРИ, 2013. – 350 с.
8. Леохин Ю.Л. Повышение качества управления мультисервисными корпоративными сетями // Телекоммуникации. – 2009. - № 7. - С. 18-21.
9. Платунова С.М. Методы проектирования фрагментов компьютерной сети – СПб: НИУ ИТМО, 2012. – 51 с.
10. Sadjadi S. J., Habibian M., Khaledi V. A multi-objective decision making approach for solving quadratic multiple response surface problems // Intern. J. Contemp. Math. Sci. 2008. V. 3, N 32. P. 1595–1606
11. John G. Kemeny, J. Laurie Snell. Mathematical models in the social sciences. – Published: New York, Blaisdell, 1963, – 145 p.
12. Леохин Ю.Л. Анализ информационной структуры корпоративной сети//Известия высших учебных заведений. Поволжский регион. Технические науки, 2008, № 4, стр. 27-40.
13. Бекасов В.Ю. Методы и модели разработки и анализа информационных структур корпоративных мультисервисных сетей. Дис. канд. тех. наук. МИЭМ, Москва, 2011.

## Development Technique of the information structure for corporate computer network

A.V. Vishnekov, E.M. Ivanova, Y.L. Leokhin

**Abstract.** The article considers the information structure choice of the corporate computer network based on the application of methods of decision support. The authors suggest a technique for calculation of the information structure parameters and evaluation criteria when making design decisions.

**Keywords:** corporate network, information structure, decision-making support methods, alternative project decision, evaluation criteria.

### References

1. Leohin Ju.L., Bekasov V.Ju. Korporativnye seti: sostojanie, perspektivy i tendencii. M.: Fond «Kachestvo», 2008. 123 p.
2. Olifer V.G., Olifer N.A. i dr. Strategicheskoe planirovanie setej masshtaba predpriyatija. M: Centr Informacionnyh Tehnologij, 2000. 680 p.
3. Olifer V.G., Olifer N.A. Komp'yuternye seti: principy, tehnologii, protokoly. 5-e izd. SPb: Izdatel'stvo «Piter», 2016. 992 p.
4. Larichev O.I. Teorija i metody prinjatija reshenij, a takzhe Hronika sobytij v Volshebnyh stranah: Uchebnik. Izd. 3-e, pererab. i dop. - M.: Logos, 2008, – 392 p.
5. Dzh. Kurouz, K. Ross. Komp'yuternye seti. Nishodjashhij podhod - M: Je, 2016. – 908 p.
6. Je. Tanenbaum, D. Ujezeroll. Komp'yuternye seti. –SPb: Piter, 2016. -955 p.
7. Uilson Je. Monitoring i analiz setej. – M: LORI, 2013. – 350 p.
8. Leohin Ju.L. Povyshenie kachestva upravlenija mul'tiservisnymi korporativnymi setjami // Telekommunikacii. – 2009. - No 7. - P. 18-21.
9. Platonova S.M. Metody proektirovanija fragmentov komp'yuternoj seti – SPb: NIU ITMO, 2012. – 51 p.
10. Sadjadi S. J., Habibian M., Khaledi V. A multi-objective decision making approach for solving quadratic multiple response surface problems // Intern. J. Contemp. Math. Sci. 2008. V. 3, No 32. P. 1595–1606
11. John G. Kemeny, J. Laurie Snell. Mathematical models in the social sciences. – Published: New York, Blaisdell, 1963, – 145 p.
12. Leohin Ju.L. Analiz informacionnoj struktury korporativnoj seti//Izvestija vysshih uchebnyh zavedenij. Povolzhskij region. Tehnicheskie nauki, 2008, No 4, p. 27-40.
13. Bekasov V.Ju. Metody i modeli razrabotki i analiza informacionnyh struktur korporativnyh mul'tiservisnyh setej. Dis. kand. teh. nauk. MIJeM, Moskva, 2011.

**Vishnekov Andrey Vladlenovich.** Academic supervisor of the NRU HSE master's programmes in the direction of "Computer Science". He graduated the Moscow Institute of electronic machine building in 1982. Professor, doctor of technical Sciences. Number of publications: more than 100. Research interests: Classification of Decision Support systems, CAD Tools, Computers and CAD Use. E-mail: avishnekov@hse.ru

**Ivanova Elena Mikhailovna.** Associate Professor of NRU HSE. She graduated Moscow Institute of electronic machine building in 1992. Candidate of technical Sciences. Number of publications: more than 50. Research interests: parallel computing, the use of mathematical methods in other fields of knowledge in the technical Sciences, classification of DSS, the teaching of computer science. E-mail: emivanova@hse.ru

**Leokhin Yuri Lvovich.** Professor of NRU HSE. He graduated Moscow Institute of electronic machine building in 1982. Doctor of technical Sciences. Number of publications: more than 100. Research interests: computer networks, control systems. E-mail: yleokhin@hse.ru