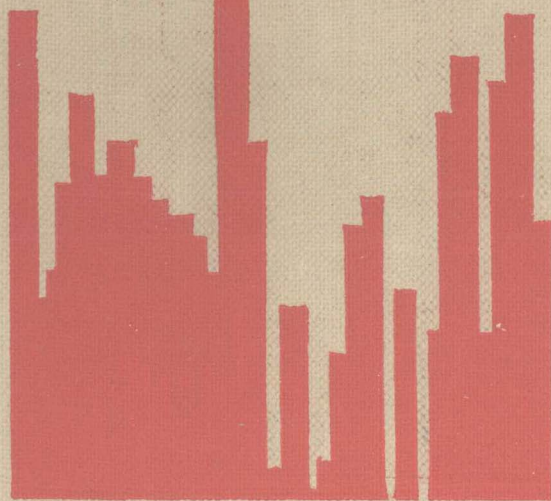


О.И. АВЕН
Н.Н. ГУРИН
Я.А. КОГАН



ОЦЕНКА
КАЧЕСТВА
И ОПТИМИЗАЦИЯ
ВЫЧИСЛИТЕЛЬНЫХ
СИСТЕМ

О. И. АВЕН
Н. Н. ГУРИН
Я. А. КОГАН

ОЦЕНКА КАЧЕСТВА И ОПТИМИЗАЦИЯ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ



МОСКВА «НАУКА»
ГЛАВНАЯ РЕДАКЦИЯ
ФИЗИКО-МАТЕМАТИЧЕСКОЙ ЛИТЕРАТУРЫ
1982

32.81

А 19

УДК 62-50

Оценка качества и оптимизация вычислительных систем.
Авен О. И., Гурин Н. Н., Коган Я. А.— М.: Наука, Главная редакция физико-математической литературы, 1982.— 464 с.

В монографии рассматривается широкий спектр вопросов, связанных с применением аналитических и экспериментальных методов для оценки качества и оптимизации проектирования и использования вычислительных систем. Большое внимание уделено конкретным применениям моделей из теории массового обслуживания, марковских моделей и методов диффузионной аппроксимации. Излагаются методы и результаты оценки качества аппаратных и программных компонент вычислительных систем в режиме нормальной эксплуатации. Рассматриваемые в книге практические примеры демонстрируют возможность повышения пропускной способности эксплуатируемых ЕС ЭВМ на 15 — 25%. Книга обобщает многолетний опыт теоретических и экспериментальных исследований авторов и отражает современное состояние проблемы.

Книга будет полезна для специалистов, связанных с проектированием, применением и эксплуатацией цифровых вычислительных систем.

Табл. 72, илл. 93, библи. 234 назв.

Олег Иванович Авен

Николай Николаевич Гурин

Яков Абрамович Коган

ОЦЕНКА КАЧЕСТВА И ОПТИМИЗАЦИЯ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

Редактор *Д. С. Фурманов*

Техн. редактор *Е. В. Морозова*

Корректоры *Т. С. Вайсберг, Л. С. Сомова*

ИБ № 11718

Сдано в набор 21.01.82. Подписано к печати 19.07.82. Т-11145. Формат 84 × 108¹/₃₂. Бумага тип. № 2. Обыкновенная гарнитура. Высокая печать. Условн. печ. л. 24,36. Уч.-изд. л. 24,86. Тираж 5000 экз. Заказ № 41. Цена 3 р. 30 к.

Издательство «Наука»

Главная редакция физико-математической литературы
117071, Москва, В-71, Ленинский проспект, 15

4-я типография издательства «Наука»

630077, Новосибирск, 77, Станиславского, 25

А 1502000000 — 109
053(02)-82 147-82.

© Издательство «Наука»,
Главная редакция
физико-математической
литературы, 1982

ОГЛАВЛЕНИЕ

Предисловие	7
Глава 1. Моделирование и экспериментальное исследование вычислительных систем	11
§ 1. Аналитические модели	11
§ 2. Имитационные модели	12
§ 3. Проблемы построения и использования моделей	14
§ 4. Экспериментальное исследование	17
§ 5. Примечания	20
Глава 2. Модели очередей в вычислительных системах	22
§ 1. Очереди в вычислительных системах	22
§ 2. Структура системы массового обслуживания	24
2.1. Входящий поток заявок (25). 2.2. Механизм обслуживания (27). 2.3. Дисциплина обслуживания (29). 2.4. Некоторые типы распределений (30). 2.5. Краткие обозначения (33). 2.6. Показатели качества (34).	
§ 3. Модели, описываемые процессами рождения и гибели	37
3.1. Процессы рождения и гибели (37). 3.2. Простейшая система $M/M/1$ (41). 3.3. Система $M/M/1$ с одноуровневым управлением (41). 3.4. Двухфазная модель вычислительной системы (43). 3.5. Система $M/M/1$ (44). 3.6. Система с квазислучайным источником заявок (44). 3.7. Простейшая модель оценки загрузки оператора вычислительной системы (47). 3.8. Модель многодоступной вычислительной системы (49).	
§ 4. Экспоненциальные системы с переменными параметрами	52
4.1. Система с гистерезисным управлением (52). 4.2. Двухфазная модель вычислительной системы с переменным режимом работы (54).	
§ 5. Системы с рекуррентным входящим потоком и рекуррентным обслуживанием	61
5.1. Цепи Маркова (62). 5.2. Обобщенный метод этапов (66). 5.3. Метод вложенных цепей Маркова (68). 5.4. Двухфазная модель вычислительной системы (71).	
§ 6. Системы с произвольным распределением длительности обслуживания, зависящим от длины очереди	74
6.1. Обслуживание с выделением единственной заявки в очереди (74). 6.2. Модель обслуживания запросов к накопителю на магнитных дисках по правилу СКАН (77). 6.3. Оценка загрузки оператора вычислительной системы (79).	
§ 7. Экспоненциальные сети массового обслуживания	82
7.1. Открытые сети (83) 7.2. Замкнутые сети (87)	

§ 8. Сетевые модели вычислительных систем	97
8.1. Сравнение различных конфигураций вычислительных систем (97). 8.2. Простейшие модели вычислительных систем с двумя устройствами ввода-вывода (103). 8.3. Модель вычислительной системы с большим набором запоминающих устройств на магнитных дисках (116).	
§ 9. Модели мультипроцессорных систем с общей памятью	132
9.1. Модели обращений в модули памяти и показатели качества (133). 9.2. Система $N \times M$ с полностью связанным интерфейсом процессоров — память (135). 9.3. Системы с односвязным интерфейсом и блоками локальной памяти (144).	
§ 10. Примечания	149
Глава 3. Марковские модели оценки качества и оптимизации	
§ 1. Необходимые сведения о конечных цепях Маркова	151
1.1. Определения (151). 1.2. Классификация состояний и цепей (152). 1.3. Регулярные цепи Маркова (153). 1.4. Поглощающие цепи Маркова (156).	
§ 2. Дискретные модели мультипроцессорных систем с общей памятью	157
2.1. Модель с равновероятными обращениями процессоров в модули памяти (158). 2.2. Модель с локализованными обращениями процессоров в модули памяти (163).	
§ 3. Оценка быстродействия конвейерных процессоров	165
3.1. Принцип конвейерной обработки (165). 3.2. Модель конвейерного процессора (166). 3.3. Факторы, снижающие быстродействие, и интерпретация понятия подчиненности (169). 3.4. Оценка влияния на быстродействие команд условного перехода (172). 3.5. Оценки быстродействия на основе марковских моделей (181).	
§ 4. Оценка качества страничных программ в виртуальной памяти	193
4.1. Модели поведения программ и показатели качества (193). 4.2. Вычисление показателей качества для марковской модели (197). 4.3. Почти разложимые модели поведения программ (200).	
§ 5. Оптимальная сегментация программ	219
5.1. Сегментация, минимизирующая среднее число межсегментных переходов (220). 5.2. Сегментация при ограничении на число межсегментных переходов (223). 5.3. Численные результаты (226). 5.4. Проверка адекватности марковской модели поведения программы (230).	
§ 6. Управление случайным множественным доступом	232
6.1. Простая тактированная система АЛОХА (233). 6.2. Система АЛОХА в случае нестойчивого протокола с проверкой несущей (239). 6.3. Приближенная декомпозиция и субпредельные распределения (240).	
§ 7. Оптимальная диспетчеризация заявок к неделимым ресурсам	242
7.1. Предварительные замечания (242). 7.2. Критерии качества и принцип оптимальности (244). 7.3. Общая схема анализа (246). 7.4. Применение общей схемы анализа (248). 7.5. Модификации общей схемы анализа (254). 7.6. Оценка стационарных вероятностей (259).	
§ 8. Примечания	260

Глава 4. Методы диффузионной аппроксимации	263
§ 1. Введение	263
§ 2. Приближения для системы $GI/G/1$	265
2.1. Определение коэффициентов сноса и диффузии (265). 2.2. Теорема сходимости (267). 2.3. Приближения для стационарных характеристик (271). 2.4. Оценка средней длины очереди к страничному диску (276).	
§ 3. Оценка загрузки центрального процессора в двухфазной модели вычислительной системы	280
3.1. Подход Гейвера — Шедлера (280). 3.2. Подход Геленбе (282).	
§ 4. Приближения для систем с переменным режимом поступления и обслуживания заявок	285
4.1. Системы с режимами, зависящими от длины очереди (285). 4.2. Система с переключаемым законом управления (288). 4.3. Система GI/GI (290). 4.4. Двухфазная модель мультипроцессорной вычислительной системы (295). 4.5. Оценка загрузки накопителя на магнитных дисках при обслуживании запросов по правилу СКАН (300). 4.6. Система с гистерезисным управлением (305). 4.7. Система, управляемая цепью Маркова (309).	
§ 5. Приближенный расчет и оптимизация систем типа АЛОХА	315
5.1. Простая тактированная система АЛОХА (315). 5.2. Система АЛОХА в случае ненастойчивого протокола с проверкой несущей (327).	
§ 6. Примечания	331
Глава 5. Экспериментальное исследование и оптимизация вычислительных систем	334
§ 1. Организация измерений динамики функционирования вычислительных систем	334
1.1. Методы измерений (334). 1.2. Измерительные мониторы (338). 1.3. Архитектура аппаратных измерительных мониторов (340). 1.4. Монитор X-RAY (344). 1.5. Принципы построения программных измерительных мониторов (347). 1.6. Системная мониторинговая программа (349). 1.7. Средство трассировки STF (351). 1.8. Монитор VM/Monitor (353). 1.9. Базовая измерительная система «Протокол» (355).	
§ 2. Экспериментальное исследование функционирования конвейерного процессора	362
2.1. Общее функциональное описание и совмещение операций в процессоре (363). 2.2. Конфликтные ситуации и экспериментальные данные (367). 2.3. Быстродействие процессора (374).	
§ 3. Измерение операционных систем пакетной обработки	379
3.1. Событийные модели процесса выполнения программ (379). 3.2. Трасса мультипрограммирования (383). 3.3. Событийные модели процессов обмена с внешними запоминающими устройствами (388). 3.4. Трасса обменов с накопителями на магнитных дисках (394).	
§ 4. Анализ операционной системы уровня J	398
4.1. Сведения об исследуемой вычислительной системе (398). 4.2. Характеристики измерительных трасс (399). 4.3. Анализ работы программ супервизора (400). 4.4. Анализ работы системных программ (405). 4.5. Активность накопителей на магнитных дисках (408). 4.6. Анализ процесса обслуживания запросов ввода-вывода (409). 4.7. Распределение обменов по программам (415). 4.8. Активность файлов (418).	

§ 5. Анализ и настройка операционной системы ОС ЕС . . .	419
5.1. Сведения об исследуемой вычислительной системе и характеристики трассы измерений (419). 5.2. Активность накопителей на магнитных дисках и наборов данных (421). 5.3. Настройка операционной системы и ее результаты (422).	
§ 6. Оптимизация использования внешних запоминающих устройств	428
6.1. Оптимизация распределения файлов по запоминающим устройствам (429). 6.2. Оптимизация размещения блоков информации на одном запоминающем устройстве (433). 6.3. Оптимизация процесса доступа к внешней памяти (444).	
§ 7. Примечания	450
Литература	452

ПРЕДИСЛОВИЕ

Под оценкой качества вычислительной системы (ВС) в настоящее время понимают получение количественных оценок, дающих представление о том, насколько хорошо ее аппаратные и программные компоненты выполняют ту работу, для которой они предназначены. Потребность в таких оценках существует, начиная с ранней стадии проектирования ВС, вплоть до ее изготовления и периода повседневной эксплуатации. Конечная цель работ по оценке качества и оптимизации вычислительной системы состоит в устранении тех недостатков, которые появляются из-за того, что ее разработчики не могут заранее предугадать все последствия принимаемых ими решений.

Интенсивные исследования по оценке качества вычислительных систем начались около 15 лет назад в связи с появлением вычислительных систем III поколения*). Особую актуальность эти исследования приобрели на протяжении последнего десятилетия, когда быстрая смена технологии, возрастающая сложность аппаратных средств и математического обеспечения, направленные на улучшение показателей стоимости и производительности, постоянно приводили к разработкам с отсутствовавшими ранее и потому трудно предсказуемыми свойствами.

*) Для одного из авторов книги, О. И. Авена, исследования в этой области начались с вопроса сугубо практического: в середине 60-х годов необходимо было выбрать техническое обеспечение ряда крупных АСУ. В дальнейшем встал вопрос об эффективном использовании имеющихся вычислительных средств. С этих работ началось формирование отдельного научного направления.

Сложность ВС, разнообразие их аппаратных и программных средств и реализуемых функций, необходимость описания и анализа процессов функционирования при различных уровнях детализации структуры объясняют существенную неравномерность развития науки об оценке качества ВС, которая при обширном потоке публикаций, посвященных решению отдельных теоретических и практических задач, весьма далека от завершенности даже по основным вопросам. Многообразие проблематики и широкий спектр теоретических и экспериментальных методов, используемых в этих публикациях, создают существенные трудности для неподготовленного читателя, желающего использовать существующие модели и методы оценки качества ВС при решении возникающих перед ним конкретных задач.

В предлагаемой книге упор сделан не на изложение многочисленных конкретных результатов, а на демонстрацию относительно небольшого числа общих подходов, при отборе которых мы руководствовались простотой и эффективностью их применения для достаточно представительного класса задач оценки качества и оптимизации ВС. Соответственно, принятая в книге схема изложения «от методов и моделей — к конкретным задачам оценки качества ВС» отличается от схемы «от конкретного класса ВС (или вычислительных устройств) — к моделям и методам», использовавшейся в ранних монографиях по теории вычислительных систем.

Книга состоит из пяти глав. Глава 1 имеет вводный характер. В ней дана общая характеристика используемых в книге моделей и методов оценки качества и оптимизации ВС. Материал последующих глав посвящен отдельным методам анализа и оптимизации ВС и может быть разбит на четыре группы достаточно автономных методов. Первую группу методов составляют методы теории массового обслуживания (глава 2), вторую — методы, основанные на использовании моделей, описываемых конечными цепями Маркова (глава 3), третью — методы диффузионной аппроксимации (глава 4) и четвертую —

методы экспериментального исследования (глава 5).

Применение рассматриваемых методов оценки качества ВС сосредоточено на относительно новых задачах, почти не освещенных в монографической литературе. Более подробно с содержанием книги можно ознакомиться, прочитав первую главу.

В книге приведена достаточно обширная библиография, однако авторы не стремились сделать ее максимально полной: где это было возможно, предпочтение отдавалось монографиям и статьям обзорного характера. Вместе с тем объединение библиографий всех цитируемых работ будет, по-видимому, уже достаточно полным.

Чтобы не загромождать изложение ссылками на литературу, большая часть из них приводится в параграфах «Примечания» в конце каждой из глав.

Текст книги разбит на главы, параграфы и пункты, причем перед номером пункта стоит номер параграфа, в котором он находится. Формулы нумеруются внутри каждого параграфа без указания его номера. Так, например, ссылка на пункт 2.3.5 означает ссылку на пункт 3.5 третьего параграфа главы 2, а ссылка на § 4.2 означает ссылку на второй параграф главы 4. Аналогично, при ссылке на формулу из другого параграфа перед номером формулы ставится номер этого параграфа, перед которым будет поставлен также номер главы, если ссылка делается не внутри одной главы. Рисунки и таблицы имеют сквозную нумерацию по всей книге.

При написании книги материал между авторами распределился следующим образом: О. И. Авен — главы 1 и 2, за исключением § 2.9; Я. А. Коган — §§ 3.1, 3.3 — 3.6 (§ 3.3 написан совместно с В. А. Сигнаевским) и глава 4; Н. Н. Гурин — §§ 5.1, 5.3, 5.4, 5.6; § 5.5 написан авторами совместно. В § 5.2 изложены результаты исследования, выполненного коллективом авторов в составе: В. С. Воробьев, В. М. Карасик, Я. А. Коган, В. А. Сигнаевский; §§ 2.9 и 3.2 написаны Л. Б. Богуславским, а § 3.7 — И. И. Бронштейном.

Для чтения книги необходимо знакомство с основами вычислительной техники и операционных систем *), а также знание высшей математики в объеме двух первых курсов втуза, включая основы теории вероятностей.

В заключение нам хочется выразить глубокую благодарность профессорам Б. М. Кагану и А. И. Никитину, которые, ознакомившись с первоначальным вариантом рукописи книги, высказали ряд полезных замечаний и рекомендаций. Особой благодарности заслуживает Г. А. Киселева, которая неоднократно перепечатывала различные варианты рукописи.

Авторы

*) Если при рассмотрении той или иной модели ВС в тексте книги не даются специальные ссылки, то это означает, что используемые сведения не выходят за рамки учебного пособия [44].

ГЛАВА 1

МОДЕЛИРОВАНИЕ И ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

Для оценки качества и оптимизации ВС применяют аналитические (математические) и имитационные (программные) модели и методы экспериментального исследования.

§ 1. Аналитические модели

Аналитическая модель ВС описывает ее функционирование посредством математических объектов. Такая модель строится, как правило, на основе понятий и символики некоторой теории, например теории массового обслуживания или теории цепей Маркова. Если выбранные показатели качества ВС удастся явно выразить в зависимости от входящих в модель параметров, то говорят, что имеет место аналитическое решение. Важное преимущество аналитических решений заключается в том, что они позволяют быстро и с минимальными затратами получить значения показателей качества в широком диапазоне изменения входных параметров модели.

Стохастический характер аналитических моделей, рассматриваемых в этой книге, отражает реально наблюдаемую случайную картину возникновения запросов на ресурсы ВС и потребления этих ресурсов. Изучаемые в последующих двух главах модели ВС формулируются в терминах теории массового обслуживания и конечных цепей Маркова.

При применении для оценки качества моделей из теории массового обслуживания ВС рассматривается как совокупность обслуживающих устройств, в качестве которых выступают различные ресурсы системы — центральный процессор, оперативная память, внешние запоминающие устройства, каналы и устройства ввода-вывода и т. д. Задания, или программы, предъявляют запросы, или заявки на обслуживание, к этим ресурсам, или об-

служивающим устройствам. Поэтому значительная часть задач оценки качества связана с анализом очередей, возникающих из-за широко используемого в ВС принципа разделения ресурсов среди многих пользователей, программ и процессов. Излагаемые в следующей главе методы теории массового обслуживания позволяют найти такие важные показатели качества, как пропускная способность, коэффициент использования и среднее время ответа.

Конечные цепи Маркова удобно применять для моделирования поведения программ. Такие модели зависят от изучаемого вычислительного процесса и используются для оценки производительности мультипроцессорных систем (§ 3.2) и конвейерных процессоров (§ 3.3), оценки качества страничных программ в виртуальной памяти (§ 3.4) и оптимальной сегментации программ (§ 3.5). Наконец, конечные цепи Маркова описывают поведение очереди к неделимому ресурсу при случайном тактированном поступлении заявок и ограниченном числе потребителей (§ 3.6). Эмпирические модели *), используемые в § 5.6 для оптимизации распределения информации на накопителях на магнитных дисках, также по существу являются марковскими. Благодаря возможности применения динамического программирования (принципа оптимальности Беллмана) марковские модели могут служить основой для синтеза оптимальных алгоритмов диспетчеризации (§§ 3.6, 3.7).

Существенно расширяют возможности исследования аналитических моделей ВС приближенные методы, среди которых наибольшее практическое применение находят рассматриваемые в главе 4 методы диффузионной аппроксимации и используемые в пп. 3.4.3 и 3.6.3 методы, основанные на приближенной декомпозиции цепей Маркова.

§ 2. Имитационные модели

Имитационная модель ВС описывает ее функционирование в виде последовательности операций или групп операций, выполняемых на ЦВМ. Составными частями

*) То есть модели, построенные на основе результатов измерений (в данном случае — процессов обмена с накопителями на магнитных дисках в режиме нормальной эксплуатации ВС).

имитационной модели являются описания элементов, образующих систему, и описание структуры системы, т. е. совокупности связей между элементами. Эти описания представляются в форме алгоритмов, на основе которых разрабатывается программа для ЦВМ. Таким образом, имитационная модель реализуется как программа, а имитационное моделирование сводится к проведению экспериментов на ЦВМ путем прогона этой программы на некотором множестве входных данных.

Имитационные модели в зависимости от используемых входных данных можно разделить на трассоориентированные и статистические. В трассоориентированной имитационной модели входные данные задаются трассой. (*Трассой* называется поток событий, имеющих место при работе ВС, которые регистрируются в хронологическом порядке, возможно, вместе с моментами их возникновения.) Трасса, применяемая для моделирования, должна достаточно хорошо отражать рабочую загрузку (типичные условия работы) ВС, причем следует иметь в виду, что на трассе сказываются не только особенности машины, но и особенности трассируемых программ. Пример использования трассоориентированной модели для оценки качества двухуровневого метода доступа к системным файлам операционной системы уровня J*) можно найти в п. 5.6.3.

В статистической имитационной модели входные данные задаются искусственно с помощью датчика случайных чисел. Поэтому параметры моделируемой рабочей загрузки можно изменять в широком диапазоне. Пример использования статистической имитационной модели для оценки точности приближенных моделей обслуживания по правилу СКАН очереди запросов к накопителю на магнитных дисках можно найти в п. 4.4.5.

Важное преимущество имитационной модели по сравнению с аналитической заключается в том, что за счет усложнения ее можно сделать сколь угодно близкой к моделируемому объекту или даже достичь полного совпадения модели с объектом. Однако часто бывает, что разработка имитационной модели занимает значительно боль-

*) Operating System J Level — операционная система, разработанная фирмой ICL для ВС ряда System 4 [96, 140]; далее в книге используется сокращение ОС J.

ше времени, а получившаяся программа требует значительно больше машинных ресурсов и дает значительно меньше информации, чем ожидалось вначале. Поэтому имитационное моделирование рекомендуется применять только в тех случаях, когда задачи оценки качества ВС столь сложны, что аналитический подход оказывается непригодным.

§ 3. Проблемы построения и использования моделей

Проблема иерархической декомпозиции. Многие трудности, встречающиеся при построении полной модели ВС, связаны с установлением иерархической структуры модели. На рис. 1 показан пример того, что мы называем иерархической структурой. Здесь модуль *A* находится на самом нижнем уровне и результаты оценки качества модуля *A* должны быть представлены в виде, легко используемом на следующем более высоком уровне *B*. Аналогично, результаты оценки качества модуля *B* используются на более высоком уровне *B* и т. д. На практике разложение ВС на модули типа показанных на рис. 1 осуществляется таким образом, чтобы время между событиями, происходящими на данном уровне, было существенно меньше промежутков времени между обращениями к более высокому уровню. При этом условии к моменту обращения к более высокому уровню модуль данного уровня войдет в режим, близкий к установившемуся, что и обеспечит упрощение анализа полной модели ВС.

При моделировании ВС во многих случаях можно ограничиться рассмотрением трех уровней *A*, *B* и *B*, соответствующих скорости работы (быстродействию) процессора (сверхоперативной или оперативной памяти), запоминающих устройств прямого доступа (ЗУПД) и пользователя, находящегося за терминалом. Выделение таких трех уровней оправдано тем, что обычно процессор успевает выполнить достаточно много команд программы между двумя последовательными обращениями к ЗУПД, а в промежутке между двумя последовательными взаимодействиями пользователя с ВС происходит достаточно много обращений к ЗУПД.

Модуль *A* с временем между событиями, измеряемым наносекундами или микросекундами, будем называть

микроуровнем, модуль *В* с временем между событиями, измеряемым миллисекундами, — *промежуточным уровнем* и модуль *В* с временем между событиями, измеряемым

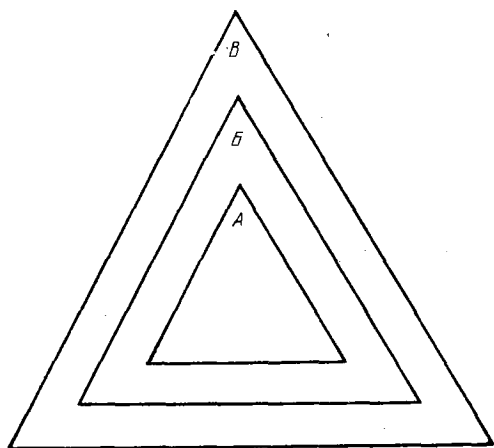


Рис. 1. Иерархия уровней оценки качества ВС.

секундами, — *макроуровнем*. Примерами моделей микроуровня являются рассматриваемые в §§ 2.9, 3.2 и 3.3 модели мультипроцессорных систем с общей памятью и модели конвейерных процессоров. Примерами моделей промежуточного уровня являются рассматриваемые в главах 2 и 4 различные модели мультипрограммирования, когда находящиеся в оперативной памяти программы (иногда называемые процессами или задачами) либо обрабатываются центральным процессором, либо находятся в состоянии обмена информацией с внешней памятью, либо ожидают обработки или обмена. На промежуточном уровне оценивается качество различных компонент операционной системы — алгоритмов распределения времени центрального процессора и планирования очереди к накопителям на магнитных дисках и алгоритмов распределения оперативной и внешней памяти. Наконец, примерами моделей макроуровня являются рассматриваемые в пп. 2.3.7, 2.6.3 и 2.3.8 модели работы оператора мультипрограммной ВС и модель многодоступной ВС.

Проблема адекватности. В процессе построения модели ВС некоторые важные зависимости могут быть пропущены и не войти в нее. В этом случае модель будет неадекватной, т. е. поведение модели для данных входных параметров будет не соответствовать поведению реальной ВС. Ясно, что неадекватная модель практически бесполезна. Если интересующий показатель качества поддается измерению, то адекватность модели можно проверить путем сопоставления фактического и модельного значений этого показателя. Если разница между ними превышает допустимые границы, то это свидетельствует о неадекватности модели. С примерами применения этого приема проверки адекватности моделей читатель может ознакомиться в §§ 2.8 и 3.5.

В случае оценки качества проектируемой ВС такая простая проверка адекватности модели не подходит. Правда, на стадии проектирования требования к точности модельных оценок, как правило, ниже и основное назначение моделей заключается в выявлении отрицательных свойств принимаемых проектных решений.

Один из способов решения проблемы адекватности заключается в построении более детальных моделей. Однако это может привести к включению в модель большого числа параметров и связей между ними, так что в целом модель станет плохо обозримой. Анализ такой модели весьма затруднителен, а имитационное моделирование требует много машинного времени. В результате модель не даст дополнительных знаний о ВС, поскольку существенные связи в ней будут теряться среди несущественных. Поэтому, как уже отмечалось выше, целесообразно иметь несколько моделей различного уровня детализации для одной и той же ВС, каждая из которых может использоваться для различных целей. Идеальная модель должна содержать в себе ровно столько детализации, сколько необходимо для той цели, для которой она строилась.

Проблемы использования аналитических моделей. Отметим некоторые трудности применения аналитических моделей. Прежде всего может оказаться, что уравнения, выражающие зависимости между параметрами системы, допускают только численное решение, для нахождения которого необходимо составить достаточно сложную программу, требующую много машинного времени. В этом