

计算机系统 性能评价

吴立德 吴霭成 编著

2
1

上海科学技术出版社

计算机系统性能评价

吴立德 吴鹰成 编著

上海科学技术出版社

责任编辑 唐仲华

计算机系统性能评价

吴立德 吴镛成 编著

上海科学技术出版社出版

(上海瑞金二路 450 号)

总发行所上海发行所发行 江苏扬中印刷厂印刷

开本 850×1156 1/32 印张 5 字数 131,000

1986 年 3 月第 1 版 1986 年 3 月第 1 次印刷

印数: 1—4,600

书号: 13119·1309 定价: 0.97 元

计算机组成原理

目 录

第一章 概论	1
§ 1.1 计算机系统的性能	1
§ 1.2 性能评价方法	2
§ 1.3 应用	3
§ 1.4 简史	5
§ 1.5 本书的目的与内容	6
第二章 计算机系统及其部件的性能指标	7
§ 2.1 部件的单项性能指标	7
§ 2.2 CPU 的综合性能指标	12
§ 2.3 系统的综合性能指标	16
第三章 测量方法	28
§ 3.1 引言	28
§ 3.2 硬件测量工具	29
§ 3.3 软件测量工具	35
§ 3.4 测量中的一些统计问题	42
第四章 选择应用	52
§ 4.1 军用计算机系统结构的选择	52
§ 4.2 程序设计语言、编译程序和支撑环境的选择	67
第五章 解析方法	71
§ 5.1 排队系统	71
§ 5.2 马尔可夫型排队模型	90

§ 5.3 其它排队模型	111
第六章 模拟方法	124
§ 6.1 引言	124
§ 6.2 随机数的生成	127
§ 6.3 负载模型和系统模型的构成	131
参考文献	155

第一章 概 论

§ 1.1 计算机系统的性能

现代的计算机系统是一个既包括硬件又包括软件的十分复杂的系统,其性能已很难用一、二个简单的指标来描述了。一般地说,它的性能应包括如下两个方面:

一个方面是它的可靠性或可利用性,亦即计算机系统能正常工作的时间,其指标可以是能持续正常工作的时间长度,如平均无故障时间 MTBF (Mean Time Between Failures),也可以是在一段时间内,能正常工作时间所占的百分比。这些当然都是综合性的指标,它们依赖于所用元、器件的可靠性;操作系统和应用软件的可靠性;系统对付有害环境的预防措施;以及一旦发生故障后,发现故障和排除故障的速度等等。

另一个方面是它的处理能力或效率。这又可分为三种类型的指标。

一类指标是各种吞吐率,如计算机系统在规定时间内能处理掉的作业个数。这类指标对主要用于批处理的计算机系统是十分有用的。一般说,它们是系统管理维护人员所十分关心的。

另一类指标是各种响应时间,即从计算机系统得到输入到给出输出之间的时间。这类指标对主要用于交互式处理的计算机系统是特别重要的。一般说,它们是计算机系统用户所最关心的。

再一类指标是各种利用率,即计算机系统(包括硬件与软件)的各种部件(如硬件中的中央处理部件,以后简记为 CPU,软件中的 FORTRAN 编译程序等)的利用率。也就是在给定的时间区

间中,被使用的时间与整个时间之比。一般说,它们是计算机系统的设计人员和管理维护人员所十分关心的。

当然上述都是一些综合性的指标,它们进一步依赖于计算机系统硬件和软件各部分的许多更具体的指标,诸如加法时间、字长、存储器容量、存取周期、编译速度等等。不仅如此,这些综合性的指标还依赖于加在系统上的工作负载。所谓工作负载指的是系统的输入,也就是用户的程序、数据、命令等。这是在讨论性能指标时的一个相当麻烦的问题。系统、负载和性能指标之间的关系如图1.1所示。

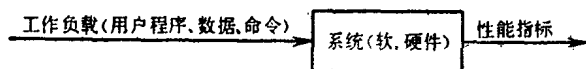


图 1.1 计算机系统和它的性能指标

除了可利用性和处理能力这两大类指标而外,可以作为计算机系统性能指标的还可以有使用的方便程度、功耗的大小、对环境的要求是否苛刻等等。

在本书中,我们主要讨论上述处理能力方面的指标,亦即主要讨论各种吞吐率、响应时间和利用率这些性能指标。它们在各种具体场合下的确切定义将在以后陆续介绍。

§ 1.2 性能评价方法

所谓性能评价就是要对指定的一个计算机系统或一类计算机系统求出其性能指标。从本世纪六十年代开始,在这方面已积累了相当多的行之有效的方法。它们大致可以分为二类。

测量方法 通过一定的测量设备或一定的测量程序(不少已是计算机系统软件中的一部分)可以直接对计算机系统测得上述各种性能指标,或与之密切有关的量,然后由它们经过一些简单的运算求出相应的性能指标。这是最直接也是最基本的方法。下面将提到的另一类方法在一定的程度上也要依赖于它。

模型方法 这时首先对要评价的计算机系统建立一个适当的

模型,然后求出模型的性能指标,以便对系统进行评价。模型中一般地包含许多参数,它们的决定往往依赖于对实际系统的测量结果。

一种常用的模型是排队网络模型。这时我们把计算机的各种部件(CPU、存储器、磁盘、输入/输出设备等,后者以后简记为 I/O 设备)当作各种“服务员”,把需要处理的各种作业当作“顾客”;而当某作业在 CPU 中处理时,就说这个作业(顾客)在接受 CPU 这个“服务员”的服务;处理完后要打印输出时,就说这个作业(顾客)在接受另一个“服务员”打印机的服务等等。从而应用排队论中的许多成果可以对计算机系统的性能指标进行计算。这种方法称为解析方法。它在模型选得适当时,可以给出相当精确的结果,而且一般说,是比较省时和省钱的。

但是,排队网络模型对系统的描述不是十分详尽的。要想更详尽地、更逼真地描述系统,就要用到模拟模型。这时模型将按照系统本身同样的方式运行。所以通过对模型进行观察和测量,并确定其性能,即可得到系统本身的性能。这种方法称为模拟方法。

模拟方法是通过一个相当大的程序来实现的,因此是比较费时和费钱的。

解析方法和模拟方法比起测量方法,有一个共同的优点是,它们不仅可以应用于现存的系统,而且也可以应用于在设计中或设想中但却尚未实现的系统,因而可以对设计工作有所帮助。

§ 1.3 应 用

性能评价的应用大体有以下三种:选择、改进和设计。

选择 当计算机用户需要为自己购买一个计算机系统时,就面临从目前市场上可买到的、众多的系统中,选择一个最适合自己的需要的系统的问题。这里的选择,包括选用什么样的机器,选用怎样的配置,要配那种操作系统,那些程序语言和应用软件包等。选

择时,自然要确定系统的性能,看看能否达到预定的要求。在这前提下,即在性能能达到预定要求的前提下,进一步选择价格便宜的,或信誉较好的等等。当然也可以反过来,在一定的价格范围中,选择性能最好的系统。因此,计算机性能评价在选择问题中是至为重要的。

改进 这是针对现有的计算机系统的。设想一个计算中心,它已有一个计算机系统。现在由于工作任务的增加,系统显得不能十分适应,而需要加以改进,即扩充与升级。例如,增加一个或多个部件(扩展它的存贮器,增加一种新的语言等等),或者用一个或多个更高性能的部件来代替现有的部件(以一个有更大容量的磁盘代替原有的磁盘,以一个较好的CPU代替原有的等等)。这里与选择时一样,也必然需要对按各种改进方案改进后的系统的性能进行评价,并从中选取改进后的性能达到预定要求,且价格便宜或费时较少等的改进方案。

设计 这里要回答的是在计算机系统设计中的问题。我们要比较几种不同的设计方案所产生的系统的性能及其价格,并从中选出性能满足要求而又价格比较便宜的设计或给定价格时性能能达到最好的设计。

不难看出,选择、改进和设计三者之间尽管有其不同的一面,例如选择问题主要是计算机系统的用户所关心的,改进问题是计算机系统的管理维护人员所关心的,而设计问题则是计算机系统的设计人员与生产制造厂所关心的等等。但也有着共同的一面,即都是要在多种系统中进行选取,其标准可以在性能满足一定要求的条件下,价格或其它指标最优,也可以是在价格等满足一定要求的条件下,性能最好。这中间,性能评价则是通过确定有关各种系统的性能而发挥自己的作用。

由此可知,评价技术为众多的人们所关心。这一点也可从1979年召开的第四届国际计算机系统模型与性能评价会议的参加者的单位分布得到证实:35%来自大学,25%来自政府的研究机构,30%来自工业界,10%来自其它。

§ 1.4 简 史

早期的计算机一则比较简单，都是单处理器，单道程序操作的，其性能基本上可用处理器的速度与存贮器的容量来刻画，二则当时的设计目标也是相当明确的：在确保一定的可靠性条件下，尽可能地快。因此，性能评价的问题并没有引起人们的注意。

直到六十年代中期，出现了多道程序的计算机系统，而且早期的这类系统的性能并不如预先设想的那么好，这才引起人们对计算机系统性能评价的兴趣，并开始了这方面的研究。

早期的一个很成功的例子是由 MIT（麻省理工学院）的 A. Sherr 在 1967 年作出的^[16]。他对当时 MIT 发展的第一台分时计算机系统 CTSS（相容的分时系统）应用有限总体的排队网络模型进行分析，得到了平均响应时间的相当精确的估计。Sherr 也因这一工作而获得 1976 年 ACM（美国计算机协会）的 Grace Murray Hopper 奖。

到七十年代，开始出现了这方面的专门的学术会议。最早的一次是 1972 年在美国 Brown 大学应用数学系召开的。会上报告的论文后来汇编成书出版了^[13]，共十九篇，分成七组：一般方法，现代系统的评价，实验设计，输入分析，回归分析，系统管理，软件可靠性。

规模和影响最大的会议要算“计算机系统模型与性能评价国际会议”。第一次是在 1974 年在法国召开的，以后大体上每一年半召开一次，至今已开了六次。会议开始时是由欧洲的一些国家与组织支持的，在 1979 年的第四次会议上决定，由国际信息处理协会(IFIP)的 7.3 工作组，即计算机系统模型工作组来负责，并且包括了欧洲以外的国家。

从七十年代中期开始，陆续出现了这方面的论文集、杂志专辑

和书籍。同时,评价中使用的许多工具也开始有了定型的产品,可以选购和使用。所有这些表明计算机系统性能评价这一计算机科学与工程中的分支的重要性日益增长,也表明这一领域正在日趋成熟。

§ 1.5 本书的目的与内容

编写本书的目的在于以较少的篇幅将计算机科学与工程中这一重要分支简要地但却系统地介绍给读者,希望能对要购买自己计算机系统的用户,计算机系统的管理维护人员以及从事计算机系统设计的工程技术人员能有所帮助,能引起他们对这一学科的进一步的兴趣。本书也可作为大学中与计算机科学与工程有关系、科开设这方面课程时的教材或教学参考书,所需学时约为 40~50 学时,相当于每周 2~3 节的一学期的课时。

阅读本书所需的预备知识为高等数学、概率统计、计算机原理、程序设计语言以及有过使用计算机的经验。但是本书前四章的大部分,特别是它的基本想法可以为具有中等文化程度和计算机一般知识的读者所接受。

全书分为六章。第一章为概论,第二章介绍刻画计算机系统各种性能的指标。顺便介绍利用现有工具书以获得各种计算机系统的某些性能指标的方法。第三章介绍测量方法,包括测量所用的硬件工具、软件工具,以及在进行测量时为保证得到有意义的数数据所应注意的一些统计方面的问题。在第四章中,我们较为详尽地介绍了美国国防部在1976年为选择军用计算机系统结构而进行的一次评价活动。这似乎也是一次规模最大的评价活动。从中可以具体地看到评价工作并非死板的,而是应结合具体的要求,综合地、灵活地运用各种方法才能获得成功的,第五章介绍解析方法,这是性能评价中当前发展得较快的一个分支。我们将简单地介绍有关排队论的一些基本概念和现有成果,以及它们在性能评价中的应用。第六章介绍模拟方法及其应用。

第二章 计算机系统及其部件 的性能指标

本章分三节。第一节介绍计算机系统各部件的各种单项指标。第二节介绍 CPU 的各种综合性指标。第三节介绍计算机系统的各种综合性指标, 特别介绍了基本的基准程序(Benchmark)的思想。

§ 2.1 部件的单项性能指标

本节介绍计算机系统各部件的单项指标, 它们一般可以从生产厂商的产品说明, 或计算机综述(Computer Review)这类工具书中查到。虽然单由这些指标还不能综合地反映出计算机系统的性能, 但通过它们确实可以对所讨论的计算机系统的性能有一个初步的、基础的了解。

一、与主机(包括 CPU 与存贮器)有关的指标

这里最基本的指标是字长、加法时间、存贮器容量、存取周期等。它们的定义如下:

字长: 基本存贮单元的长度, 其单位为比特(bit)数。一般的微型计算机, 其字长为 8 比特或 16 比特; 小型计算机为 16 比特和 32 比特; 大型计算机的字长则有 8、16、24、32、36、48、60、64、72、128、256 比特等多种, 据 1981 年 Computer Review 的统计, 其中约有 40% 是 32 比特或 36 比特, 另有 46% 是 8 比特或 16 比特, 而超过 36 比特的只占 10% 不到, 8 比特字长机器的比例很高这一点是跟 IBM (国际商用机器) 公司生产大量 8 比特的大型机有关。

存贮器容量：存贮器的大小，其单位一般为千字节(KB)或百万字节(MB)。一字节是指八个比特，即八个二进位。一般的微型机和小型机，其存贮器容量在1 MB 以下。而据1981年 Computer Review 的统计，大型机中近一半的存贮器容量在1 MB~4 MB 之间，个别也有高达64MB 的。

存取周期：从存贮器中读出一个字或将一个字写入存贮器所需的时间，其单位为微秒(μs)或毫微秒(ns)。它提供了计算机处理速度的一个粗糙的估计。对不同的计算机，其变化范围是相当大的，可以从几十微秒到几个毫微秒。就目前的大型机而言，据1981年 Computer Review 的统计，其平均存取周期约为600 ns，约有25% 是在500 ns~800 ns 之间，也有11% 是在100 ns 以下。

加法时间：执行定点加法所需的时间，其单位为微秒或毫微秒。对不同的机器，其变化范围是很大的，可以从几十微秒到几个毫微秒。

除上述这些指标外，还有许多其它的指标，它们是：

是否有高速缓冲存贮器，如有的话，它的容量和存取周期怎样？为了提高处理速度，近来一半以上的大型计算机都配备了高速缓冲存贮器，其容量也在不断扩大，有的已达到256 KB；据1981年 Computer Review 的统计，其容量一般在8~16 KB 之间，小于8 KB 的和大于32 KB 的都约占25%。

有多少指令？例如是否有硬件实施的乘、除法指令，浮点运算指令，十进制运算指令，以及双倍位乃至多倍位的运算指令等等。

有几种寻址方式，例如是否有间接寻址的能力？是否有基地址重定位，即有否通过基地址寄存器将存贮器中的程序放入存贮器中另一区域执行的能力等等。

是否有虚存，即是否有将虚地址自动转换成物理地址的能力。

是否有专用的中断寄存器，以便 OPU 可对不同的中断直接转

到相应的中断处理程序入口进行处理？是否有响应多种优先级中断的能力等等。

可否允许多处理器结构，特别是否有某种类型的“测试和置位”指令以便实施同步功能。

可以接多少外部设备和终端，与外部设备之间的最大传输速率为多少等等。

存贮器有否校验措施和保护措施，例如是否有一位奇偶校验位以便能发现单个出错，或是有若干个校验位以便不仅能发现而且能自动地校正多个出错。又如存贮器中是否有一特定的区域，对这一区域用户程序只能读出但不能写入。以及更一般地，是否用户程序总是无法破坏系统程序和其它用户程序等等。

二、与外部设备有关的指标

1. 磁盘 与磁盘有关的指标包括类型、容量、平均存取时间、传输速率等。

目前常用的磁盘有如下数种：固定磁头的磁盘，活动磁头的磁盘，可装卸的磁盘以及 Winchester 磁盘等数种。Winchester 磁盘由于采用了所谓的 Winchester 技术，因而有比一般磁盘高得多的记录密度。

固定磁头的磁盘，它的每一个磁道都有一个磁头，因此它的平均存取时间就是磁盘旋转一周的时间的一半，因而是比较快的，典型的数值为 10~30 毫秒。传输速率则是指磁头已找到正确位置后开始与主机交换数据的速率，它依赖于磁盘旋转的速度和磁盘的记录密度。例如 IBM3810-B2 型磁盘就是一种固定头磁盘，其容量为 64MB，平均存取时间为 27 毫秒，传输速率为 1031 KB/s。

活动磁头磁盘，它的每一个磁面才有一个磁头，因此它的平均存取时间包括了两部分，一是磁头移动到所要读写的磁道上的时间，二是等待磁道上要读写的数据旋转到磁头下的时间，后者就是固定磁头磁盘的存取时间。因而它要比固定磁头的磁盘为慢。典型的存取时间为 30~60 ms。例如 IBM3333-1 就是活动磁头的磁盘，其容量为 100 MB，平均存取时间为 60 ms，传输速率为 312 KB/s。

可装卸的磁盘是活动磁头的，其特点为整个磁盘是装在一个盒子中的，可以取下和装上，象磁带那样，因而有其方便之处。例如 IBM5444-A3 就是这种类型的，其容量为 2 MB，平均存取时间为 126 ms，而传输速率为 199 KB/s。

与磁盘有关的另一些指标是，每个控制器可带几个驱动器和每个驱动器可带几个磁盘等。

2. 磁带 目前常用的是半时带，有 7 道和 9 道两种格式。常用的记录密度为每吋 200、556、800、1600、6250 比特等几种，特别是 800 和 1600 两种。带速则有相当大的变动，可从每秒 2 吋到每秒 125 吋，相应地数据传输速率也有相当大的变动。

每盘带的总长度一般约为 2400 英尺，因而其容量对 9 道磁带来说，视不同的记录密度而在 5.76 MB 到 180 MB 之间。

例如 IBM3410-1 磁带，它既能用于 7 道格式也能用于 9 道格式，记录密度可用每吋 200、556、800 比特中的任一种，带速为每秒 12.5 吋，传输速率最高为 20 KB/s。

3. 行式打印机 目前有多种类型的行式打印机：机械打击式的、静电的、热敏的、喷射式的、以及光敏的等。所打印的字体则有实体的与点阵式的两种。

能打印的字符集也有所不同，从只能打 48 个不同字符，到能打 64 个、96 个、以至 128 个字符。有些打印机还能打印 APL 语言所需的各种特殊字符。

其它重要的指标是：每行的字符数，常见的有每行 80、96、120、132、136 个字符等；打印速度，即每分钟能打印的行数，其变动是比较大的，从每分几十行到高达每分几万行的都有。

4. 其它 除磁盘、磁带、行式打印机外，常见的外部设备尚有终端、卡片输入机、纸带输入机、串行打印机、纸带穿孔机、绘图仪、图形显示等等，这里就不再一一介绍了。

三、配备的软件

现在的计算机，其使用的效率，不仅取决于它的硬件（包括主机和外部设备），同时也取决于它所配备软件的多少以及这些软件

的质量。

下面是一些最重要的软件,一般说,配备得越多越好。

汇编

宏汇编 除了与机器指令直接对应的汇编指令外,尚有宏指令,这是界于汇编语言与程序设计语言之间的一种语言。

实时监控程序 由非时序的外来事件驱动的一种操作系统。

分时监控程序 一种执行程序,由它们来实现多个联机用户同时使用一台计算机进行工作。每个用户通过控制台或终端采用问答方式控制自己的作业运行。系统则把 CPU 时间轮流分配给各联机的作业,每个作业只运行极短的一个时间片,如果在时间片结束之前,作业尚未完成,它就被时钟中断,等待下一轮再运行,而 CPU 则转去处理下一个联机用户的作业。这样,各用户的每次要求都能得到快速的响应,给用户以他独占计算机的印象。

批处理监控程序 一种执行程序或管理程序,它把一批作业输入到磁带上,然后不间断地逐个地(即顺序地)调入处理器进行运行,从而实现了作业间转化的自动化,缩短了作业建立时间和操作时间。

数据库系统 这是在操作系统与应用程序之间的中介程序设计系统,它使用户可以查询和处理数据文件。

通讯软件 这是一个软件包,它使系统可以通过专用的或公用的通讯线路把自己变为一个分布式系统或一个计算机网络的一个组成部分。

各种程序设计语言的解释程序或编译程序,最常见的为:

APL: 一种特别适合于数组运算的程序设计语言,所用的键盘上要有专用的符号键。

ALGOL: 一种面向科学计算的程序设计语言,在欧洲较流行。

BASIC: 一种最简单的程序设计语言,特别适用于初学者。

COBOL: 一种面向商业数据处理的语言。

FORTTRAN: 另一种面向科学计算的程序设计语言,也是目

前使用得最广泛的一种程序设计语言。

PL/1: 一种有力的既适合科学计算又适合商用数据处理的设计语言。

RPG: 一种生成指定格式报告的程序设计系统。

PASCAL: 一种面向过程的结构程序设计语言, 它是七十年代初期在 ALGOL 的基础上发展起来的。

§ 2.2 CPU 的综合性能指标

由于在相当长的一段时间中, CPU 被认为是计算机系统中最主要的资源, 因而有着不少关于它的性能的研究。又因为 CPU 是用于处理数据的, 所以它的性能应由它处理数据的能力来刻画。本节中将介绍一些这方面的综合指标。

一、Gibson 混合与 MIPS

由于计算机是通过执行一系列的指令来实现其处理数据的任务的, 因此指令的执行时间或其倒数——单位时间中执行的指令条数就可以用来作为刻画 CPU 的性能的指标。

但由于不同的指令的执行时间是不同的, 所以一个自然的想法是用指令的平均执行时间, 或用单位时间中平均执行的指令条数来作为刻画 CPU 的性能的综合指标。

所谓指令的平均执行时间就是各种不同指令执行时间的某种加权平均值。由于对给定的计算机, 它的各种指令的执行时间是知道的, 因此只要确定了各种指令的权后就可以求出相应的指令的平均执行时间, 而一种很自然的权就是各种指令出现的频率。

这种方法最早由 Gibson 在五十年代提出。他当时针对 IBM704 计算机, 根据一组有代表性的 FORTRAN 程序, 统计了各类指令出现频率, 并用这些频率作为上述加权平均中的权, 而求得指令的平均执行时间和每秒平均执行的指令条数, 作为 CPU 的性能的综合指标。

在 Gibson 之后, 还有不少人针对其它的计算机, 根据有不同