

软件生产的优化设计

王培永 赵越 (浙江省计算技术研究所)

摘 要

This paper defines the production function and cost function of software production from the point of view of economics. According to the two functions, an optimal plan model for software production is proposed, and the meanings and the estimating methods of the parameters in the model are discussed. Finally, a multiobjective optimal plan model is given.

随着计算机应用的日益普及和深化, 各类计算机软件的数量正以惊人的速度急剧增加, 且现代软件的模式也往往十分庞大。例如, 包含数百万行代码、耗资几十亿美元、花费几千人年工作量的软件产品在工业发达国家已经屡见不鲜。

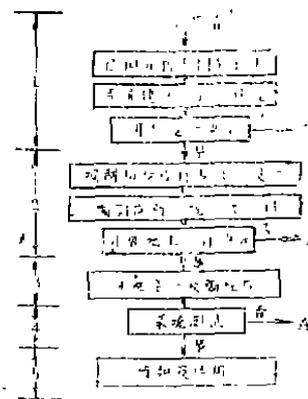
由于微电子学技术的进步, 计算机硬件成本一般每五年下降两至三个数量级, 且硬件质量稳步提高; 与此同时, 计算机软件成本却在逐年上升, 软件质量得不到可靠的保证, 软件开发的生产率也远远不能满足普及计算机应用的要求。西方计算机科学家把在软件开发和维护过程中遇到的一系列严重问题统称为“软件危机”。其中最主要的问题之一就是软件开发成本和进度的估计很不准确, 而软件成本在计算机系统总成本中所占比例的逐年上升, 更加剧了这一问题的严重性。因此, 降低软件开发和维护成本目前已成为降低整个系统总成本的关键。我们认为必须从根本上改变目前对软件生产规划和安排的盲目性, 必须科学地对软件生产进行分析, 进而予以优化。

一、生产函数和成本函数

软件工程学研究的是: 如何应用科学理论和工程技术来指导软件的开发, 从而达到以较少的投资获得高质量软件的最终目的。

在软件工程学中, 一般把软件生命周期划分成五个阶段: 分析、设计、编程、测试和运行, 它们之间的相互关系及各阶段的参与者如图1所示。

我们可把软件生产的每一阶段视为流水作业的一道工序, 即每个阶段构成一个相对独立的子生产

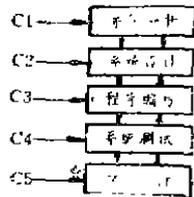


1. 系统分析及可行性研究阶段; 参与者: 用户、高级程序员。
 2. 系统设计阶段; 参与者: 高级程序员。
 3. 编程阶段; 参与者: 高级、初级程序员。
 4. 测试阶段; 参与者: 另一独立部门。
 5. 运行阶段; 参与者: 用户、操作人员。
- A: 根据各阶段的结果决定研制过程是否需要返回至某阶段。

图1软件生产诸阶段示意图

过程。就每一子生产过程而言, 都存在一种“投入”(包括时间、经费等), 而其“产出”则正是下一阶段所需的文档资料或程序。为了便于下面的进一步讨论, 我们可把图1中描述的复杂的软件生产过程抽象为如图2所示的较简明的过程。

根据经济学的投入产出理论, 我们可定



其中, C1、C2、C3、C4、C5为各阶段的投入

图2 五个阶段投入和产生的简单关系

义软件生产的生产函数为:

$$D=f(f_1, f_2, \dots, f_n) \quad (2-1)$$

其中, D 是生产过程的产出, f_1, f_2, \dots, f_n 是生产因素, 包括

- ①硬件、软件(操作系统及通信网等);
- ②分析员、程序员及管理人员;
- ③进度。

软件生产过程的产出 D 应是适当的文档资料或程序, 但对它们进行度量却比较困难。因此我们采用变通方法, 以整个系统的规模及复杂度作为产出的度量。另外, 不妨假设在某一时期内, 计算机软硬件及通信网络等的性能与成本以及人员工资等有关生产因素基本不变, 即可将这些因素视为“技术常数”。那么上述生产函数(2-1)可定义为:

$$D=AP^\alpha S^\beta H \quad (2-2)$$

式中, D 为所要求的产出, P 为人员数, S 表示开发进度, H 为开发环境的投资额, α 、 β 和 γ 为小于1的常数, A 为软件开发单位的生产效益系数。

由此, 软件生产的成本函数可定义为:

$$C=P_p \cdot P \cdot S + P_s \cdot S + H \quad (2-3)$$

其中, C 为成本, P_p 为单位“人员时间”的价格, P_s 为单位“时间”的价格。

如果我们把软件生产划分为 k 个阶段, 则各阶段的生产函数为:

$$\delta_i D = A_i P(i)^{\alpha_i} S(i)^{\beta_i} H(i)^{\gamma_i}$$

$$i=1, 2, \dots, k$$

其中, δ_i 为第 i 阶段占整个系统开发的比例, $\delta_i > 0$, $i=1, 2, \dots, k$, 且有 $\sum_{i=1}^k \delta_i = 1$ 。

A_i 、 $P(i)$ 、 $S(i)$ 、 $H(i)$ 、 α_i 、 β_i 、 γ_i 等为第 i 阶段的相应变量或常系数。

与上述生产函数相类似, 各阶段的成本函数为:

$$C_i = P_p(i)P(i)S(i) + P_s(i)S(i) + H(i) \quad i=1, 2, \dots, k$$

显然, 整个软件生产过程的生产函数和成本函数分别为:

$$D = \sum_{i=1}^k A_i P(i)^{\alpha_i} S(i)^{\beta_i} H(i)^{\gamma_i} \quad (2-4)$$

$$C = \sum_{i=1}^k [P_p(i)P(i)S(i) + P_s(i)S(i) + H(i)] \quad (2-5)$$

如前所述, 软件生产过程一般可划分为五个阶段, 则有 $k=5$ 。

二、单目标优化设计模型

软件工程需要解决的一个问题是如何在条件允许的范围内以尽可能少的投资获得高质量的软件, 根据上一节所述的生产函数(2-4)和成本函数(2-5), 我们提出采用下述非线性最优化设计模型来解决上述问题:

$$\text{Min} \sum_{i=1}^k [P_p(i)P(i)S(i) + P_s(i)S(i) + H(i)]$$

$$\text{S.t. } \delta_i D = A_i P(i)^{\alpha_i} S(i)^{\beta_i} H(i)^{\gamma_i}$$

$$\sum_{i=1}^k H(i) \leq H_0 \quad i=1, 2, \dots, k$$

$$\sum_{i=1}^k S(i) \leq S_0$$

$$\sum_{i=1}^k P(i) \leq P_0 \quad (3-1)$$

关于模型(3-1)中诸参数的涵义详细讨论如下:

$P_p(i)$: 第 i 阶段单位“人员时间”的价格。一般各个阶段的 P_p 值不尽相同, 例如, 设计阶段的参与者一般为高级程序员, 则相应的单位“人员时间”的价格自然比其它各阶段的高一些。

$P_s(i)$: 第 i 阶段单位“时间”的价格。一般来说, 各阶段的时间价格几乎一样, 但有时某一阶段

对时间的要求可能非常高, 则其时间价格就应上升, 并且随着时间紧迫程度的增加, P_i 也应按比例增值。

δ_i : 第 i 阶段在整个软件生产周期所占的比例。
一般在阶段划分固定的情况下, δ_i 基本固定且大于零, 并有

$$\sum_{i=1}^k \delta_i = 1.$$

A_i : 软件生产单位关于第 i 阶段的生产效率的综合评价系数。它能反映出特定单位关于第 i 阶段的生产水准, 也可以说是参数 α_i 、 β_i 和 γ_i 的综合体。

α_i : 组织管理能力系数。在一般情况下, $\alpha_i < 1$ 。如果某单位的组织管理能力较强, 则 α_i 就较大; 即在同样条件下, 若增加所投入的生产人员, 则 α_i 较大的单位的产出比 α_i 较小的单位的产出增加得多。

β_i : 人员素质系数。 β_i 与软件生产单位人员素质的高低成正比, 一般有 $\beta_i < 1$ 。

γ_i : 投资效益系数。 γ_i 与软件生产单位投资效益的好坏成正比, 同样, $\gamma_i < 1$ 。

H_0 : 开发环境投资限制。

S^0 : 时间限制。

P_0 : 人员限制。

参数 $P(i)$ 、 $S(i)$ 和 $H(i)$ 的涵义已在上一节中讨论, 在此不再赘述。

模型(3-1)很有用处。例如, 我们可用它分析技术不发达的手工业作坊的时期, 它往往以单干的生产形式为主的原因。在那时, 由于技术落后, 生产环境长期得不到改善, 故生产环境投资基本固定, 并且时间价格几乎为零, 生产进度没有限制。由此, (3-1) 可简化为:

$$\text{Min} \sum_{i=1}^k P_p(i) P(i) S(i)$$

$$\text{s.t. } \delta_i D = A_i P(i)^{\alpha_i} S(i)^{\beta_i}; \quad i=1, 2, \dots, k$$

显然, $P(i)=1, i=1, 2, \dots, k$ 为上述优化问题的解, 即单干的生产形式是那时的最优生产模式。

但在科技高度发达的信息时代, 时间越来越宝贵, 其价格越来越高。另外, 许多软件的规模已达到几千甚至上万人年, 一个人一生全力以赴也不可能完成这样大的软件系

统, 这就需要数千或更多的软件人员进行合作才有可能完成。所以, 必须合理地组织人员, 制定计划, 从而以最经济、最快的进度高质量地完成软件开发任务。

在进行软件生产时, 欲通过运用模型(3-1)来达到优化设计目的的前提是所给的参数必须正确。对于具体的软件生产单位, 我们可以根据该软件生产的历史数据来正确地估算参数 δ_i 、 A_i 、 α_i 、 β_i 和 γ_i 。具体估算方法如下: 设在某一相对稳定时期, 某软件生产单位完成了 m 个软件生产任务, 这里要求 $m \geq k$ 。我们可通过解下述无约束极值问题来求出参数 δ_i^* 、 A_i^* 、 α_i^* 、 β_i^* 和 γ_i^* 的值:

$$\text{Min} \sum_{i=1}^m \sum_{j=1}^k \left[\delta_i^* D_j - A_i^* P_j(i)^{\alpha_i} S_j(i)^{\beta_i} H_j(i)^{\gamma_i} \right]^2 \quad (3-2)$$

其中, D_j 、 $P_j(i)$ 、 $S_j(i)$ 和 $H_j(i)$, $j=1, 2, \dots, k$; $j=1, 2, \dots, m$ 为已知的历史数据。上述求得的参数 δ_i^* 、 A_i^* 、 α_i^* 、 β_i^* 及 γ_i^* 除可用于进行优化设计外, 还可用于评估该生产单位的生产率水准, 分析各阶段的生产状况, 并有助于改进各环节的不足之处, 提高生产效益, 进而为设计出更佳的优化设计方案创造条件。

三、多目标优化设计模型

在上一节, 我们建立并讨论了单目标优化设计模型(3-1), 本节我们将建立并讨论多目标优化设计模型。

目前, 随着计算机科学在理论和技术上的飞速发展, 软件的生产周期及使用周期日益缩短。这就要求我们在尽可能降低软件生产成本的同时, 以最快的速度生产出高质量的软件, 即优化目标有两个: 尽可能降低成本和尽可能加快速度。欲同时实现这两个目标, 应采用下述优化设计模型:

$$\text{Min} \sum_{i=1}^k S(i)$$

$$\begin{aligned} & \text{Min} \sum_{i=1}^k \left[P_p(i)P(i)S(i) + P_s(i)S(i) + \right. \\ & \qquad \qquad \qquad \left. + H(i) \right] \\ \text{s.t.} & \delta_i D = A_i P(i)^\alpha \cdot S(i)^\beta \cdot H(i)^\gamma, \\ & \qquad \qquad \qquad i=1, 2, \dots, k \\ & \left\{ \begin{aligned} & \sum_{i=1}^k H(i) \leq H_0 \\ & \sum_{i=1}^k S(i) \leq S_0 \\ & \sum_{i=1}^k P(i) \leq P_0 \end{aligned} \right. \quad (4-1) \end{aligned}$$

除了上述两个优化目标外，关于软件设计的优化目标还可能有两个：尽可能减少所投入的软件开发人员和尽可能降低对开发环境的投资。因此，我们可以建立一般的多目标优化模型如下：

$$\begin{aligned} & \text{Min} \sum_{i=1}^k P(i) \\ & \text{Min} \sum_{i=1}^k S(i) \\ & \text{Min} \sum_{i=1}^k H(i) \\ & \text{Min} \sum_{i=1}^k [P_p(i)P(i)S(i) + P_s(i)S(i) + \\ & \qquad \qquad \qquad H(i)] \\ \text{s.t.} & \delta_i D = A_i P(i)^\alpha \cdot S(i)^\beta \cdot H(i)^\gamma, \\ & \qquad \qquad \qquad i=1, 2, \dots, k \\ & \left\{ \begin{aligned} & \sum_{i=1}^k H(i) \leq H_0 \\ & \sum_{i=1}^k S(i) \leq S_0 \\ & \sum_{i=1}^k P(i) \leq P_0 \end{aligned} \right. \quad (4-2) \end{aligned}$$

在具体应用模型 (4-2) 时，设计者可根据具体的开发情况对该模型进行灵活的变换

和组合。例如，可以采用对各目标加权求和的方法进行优化设计：

$$\begin{aligned} & \text{Min} \left\{ \lambda_1 \sum_{i=1}^k P(i) + \lambda_2 \sum_{i=1}^k S(i) + \lambda_3 \sum_{i=1}^k H(i) \right. \\ & \qquad \qquad \qquad \left. + \lambda_4 \sum_{i=1}^k [P_p(i)P(i)S(i) + P_s(i)S(i) + \right. \\ & \qquad \qquad \qquad \left. H(i)] \right\} \\ \text{s.t.} & \delta_i D = A_i P(i)^\alpha \cdot S(i)^\beta \cdot H(i)^\gamma, \\ & \qquad \qquad \qquad i=1, 2, \dots, k \\ & \left\{ \begin{aligned} & \sum_{i=1}^k H(i) \leq H_0 \\ & \sum_{i=1}^k S(i) \leq S_0 \\ & \sum_{i=1}^k P(i) \leq P_0 \end{aligned} \right. \quad (4-3) \end{aligned}$$

其中， λ_1 、 λ_2 、 λ_3 和 λ_4 分别为四个优化目标的技术权因子， $\lambda_i \geq 0$ ($i=1, 2, 3, 4$)，并且 $\sum_{i=1}^4 \lambda_i = 1$ 。特别地，当 $\lambda_1 = \lambda_2 = \lambda_3 = 0$ 时，模型 (4-3) 即为模型 (3-1)；当 $\lambda_1 = \lambda_3 = 0$ 时，模型 (4-3) 即为模型 (4-1) 的目标加权求和形式。

模型 (4-3) 是模型 (4-2) 的一种具体应用形式，而模型 (4-2) 还可以有别的应用形式。因此，在具体应用模型 (4-2) 进行优化设计时，设计者应认真选择最适合其特定情况的应用形式。当然，有时也可数种形式并用，以求事半功倍之效。

四、实际运用

在这一节，我们将结合本单位的具体开发数据介绍优化设计模型的实际运用情况。由于在不同的软件生产单位之间具体的生产条件、管理措施和开发方法差别很大，故下面所给的许多参数值不一定能适用于其它单位，特请读者注意。

目前，我国的计算机应用大多集中在信息管理、过程控制、办公室自动化等方面，这类系统的投资规模一般不太大，故在开发时基本上是利用已有的开发环境。因此，对于大部分软件生产机构来说，关于开发环境的投资可以不列入其软件生产的优化目

标之中。另一方面, 由于其软件生产标准化尚未普及, 且软件生产各开发阶段也未作明确的划分, 所以, 我们可把整个生产过程视为一个阶段, 即 $K=1$ 。基于上述情况, 我们可相应地定义生产函数和成本函数如下:

$$D = AP^{\alpha}S^{\beta}$$

$$C = P_1PS + P_2S$$

而模型 (4-3) 则可简化为:

$$\text{Min}[\lambda_1 P + \lambda_2 S + \lambda_3 (P_1PS + P_2S)]$$

$$\begin{aligned} \text{s.t. } & D = AP^{\alpha}S^{\beta} \\ & P \leq P_0 \\ & S \leq S_0 \end{aligned} \quad (5-1)$$

还有, 我们可以通过解下述无约束极值问题来估计参数 A 、 α 和 β 的值:

$$\text{Min} \sum_{i=1}^m (D - A^* P_i^{\alpha} S_i^{\beta})^2 \quad (5-2)$$

例如, 以我们研究所近几年来所开发的 17 个软件系统的实际完成情况为样本 ($m=17$), 用 (5-2) 式可求得: $A^*=0.24$, $\alpha^*=0.7$, $\beta^*=0.8$ 。故相应的生产函数为:

$$D = 0.24 P^{0.7} S^{0.8} \quad (5-3)$$

式中, P 以人数为单位的, S 以月为单位的, D 以人年为单位的。

有了 (5-3) 式后, 就可对软件生产进行科学的预计、规划和安排。例如, 对于一个工作量为 20 人年的软件系统, 若投入 C 个人进行开发, 则所需的开发时间为:

$$S = \left(\frac{20}{0.24 \times C^{0.7}} \right)^{1/0.8} = 52.5 \text{ (个月)}$$

而若投入 10 个人, 则为

$$S = \left(\frac{20}{0.24 \times 10^{0.7}} \right)^{1/0.8} = 33.6 \text{ (个月)}$$

欲在两年内完成该项目, 则所需投入的人数为:

$$P = \left(\frac{20}{0.24 \times 24^{0.5}} \right)^{1/0.7} = 14.7 \text{ (人)}$$

上述例子中的结果与我们单位目前的实际生产情况是比较吻合的, 因此, 我们认为只要合理地确定有关参数的值, 就能用前面所述的各种优化设计模型设计出所期望的最佳软件生产方案。

五、结论

根据目前软件生产的发展趋势, 对软件生产进行优化设计势在必行, 本文提出单目标和多目标优化设计模型正是为了满足这一需要。但在具体运用

这些模型时, 设计者应根据其软件生产单位的特点及生产规模, 对所选用的模型进行合理的变换和组合, 并采用相应的参数估算方法正确地估算各参数的值, 各参数值的正确性依赖于所收集的历史数据的准确性和代表性。当然, 这些参数还应在实际应用中逐步加以校正, 从而设计出能适合特定单位的最优软件生产方案。

参考文献

- [1] A. 乔伊科奇, D. R. 汉森, L. 达克斯丁, “多目标决策分析及其在工程和经济中的应用” (中译文), 航空工业出版社, 1987年。
- [2] Martin L. Shooman, “Software Engineering: Design Reliability and Management”, McGraw-Hill Book Co., 1983.
- [3] William S. Davis, “Systems Analysis and Design—A Structured Approach”, Addison-Wesley Publishing Co., 1983.
- [4] Barry W. Boehm, “Software Engineering Economics”, Prentice-Hall Inc., 1981.
- [5] 张海藩, “软件工程学导论”, 清华大学出版社。
- [6] 赵越, “软件工程方法在开发实际系统中的应用”, 浙江大学学报, 1988年, 第2期, 第22卷。
- [7] 陈火旺, 祁润年, “软件工程学: 过去、现在与将来”, 计算机工程与科学, 1987年, 第2期。
- [8] M. M. Manai and T. J. Teorey, “Cost/Benefit Analysis for Incorporating Human Factors in the Software Lifecycle”, Commun. of the ACM, Vol. 31, No. 4, 1988.
- [9] H.D. Mills, “Software Development”, IEEE Trans. of Software Eng., Vol. SE-2, No. 4, 1976.
- [10] A. J. Symonds, “Creating a Software Engineering Knowledge Base”, IEEE Software, Vol. 5, No. 2, 1983.
- [11] C. R. Vick and C. V. Ramamoorthy, “Hand-book of Software Engineering”, Van Nostrand Reinhold, N. Y., 1984.
- [12] P. W. Metzger, “Managing a Programming Project”, Prentice-Hall, Inc., 1981.