基于模糊理论的软件开发成本估算

任永昌[1,2] 邢 涛[3] 刘大成[4]
(渤海大学信息科学与工程学院 锦州 121013)[1] (江南计算技术研究所 无锡 214083)[2]
(北京城市系统工程研究中心 北京 100089)[3] (清华大学工业工程系 北京 100084)[4]

摘 要 针对不确定性对成本估算的影响,提出了运用模糊理论来估算软件开发成本。通过对模糊理论估算公式的原理描述与数学推导,创建了数学模型,给出了确定隶属函数和计算贴近度的方法;论述了数据库规模与软件规模的关系,分析了影响数据库规模的因素,设计了数据库规模估算的数学模型;指出了运用模糊理论估算软件开发成本的步骤;对模糊理论方法进行了总结。运用模糊理论估算软件开发成本,能减少因素变化的不确定性引起的估算用时较长、估算过程复杂、估算不准确等问题。

关键词 模糊理论,软件开发成本估算,隶属函数,数据库规模估算
中图法分类号 TP311 **文献标识码** A

Software Development Cost Estimates Based on Fuzzy Theory

REN Yong-chang[1,2] XING Tao[3] LIU Da-cheng[4]
(College of Information Science and Engineering, Bohai University, Jinzhou 121013, China)[1]
(Jiangnan Institute of Computing Technology, Wuxi 214083, China)[2]
(Beijing Research Center of Urban Systems Engineering, Beijing 100089, China)[3]
(Department of Industrial Engineering, Tsinghua University, Beijing 100084, China)[4]

Abstract Aiming at the impact of the uncertainly cost estimates, we presented using fuzzy theory to estimate software development cost. By describing the principle of estimate formula of fuzzy theory and Mathematical Derivation, we created a mathematics model using fuzzy theory to estimate software development cost, gave the method of determining the membership function and computing similarity measure, discussed the relationship of database scale and software scale, analyzed the impact factor of database scale, designed the mathematics model of database scale estimation, pointed out the step of software development cost estimates using fuzzy theory. The summary of fuzzy theory method using fuzzy theory method to estimate the software cost can reduce the time cost of estimates, the complex of processing and the inaccurate of estimates which are caused by the variety of uncertain factors.

Keywords Fuzzy theory, Software development cose estimates, Membership function, Database scale estimation

软件成本估算是软件工程领域重要的研究方向,是可行性研究阶段的重要任务,也是制定计划的依据。随着软件项目规模的日益增大和投资的不断增加,软件成本估算的重要性更加突出。当前软件成本估算不准确的原因之一就是没有解决不确定性对成本估算的影响。软件成本估算是一种事先行为,是在假设条件下估算的结果。但用户的需求、企业高层的要求、市场的需要、政策法律的变化等,都可能导致设计和计划的变更,从而影响成本估算。在很多领域,解决不确定性问题是运用数学方法,将不确定性、模糊性问题进行数字化或归一化处理,从而转化成确定性问题进行求解。本文运用模糊理论估算软件开发成本,解决了不确定性对成本估算的影响。

1 数学模型的建立

1.1 估算原理

软件开发成本估算,是对成本的预先估计或预测。不存在两个完全相同的软件项目,总会有差异,但众多的软件项目中存在相似性,不同的软件项目相似性也不同。估算的基本原理建立在软件项目之间的相似性上。

对于某个要估算的软件项目(简称预估项目),从数目众多的已经知道软件开发成本的项目(简称典型项目)中,找出与之最相似的若干个软件项目(简称相似项目),利用相似项目的实际开发成本,采用指数平滑法进行估算,这就是成本估算的基本原理。

在预测研究中越近期的数据越受到重视,时间序列数据中各数据的重要程度由近及远呈指数规律递减,故对时间序列数据的平滑处理应采用加权平均方法。指数平滑法就是一种加权平均法,但这个权数是根据过去的预测数和实际数的差异确定的,这样取得的权数称为平滑系数。可以利用该系

到稿日期:2009-11-10 返修日期:2010-01-29 本文受国家自然科学基金(70871067),国家863计划(2005AA501560),辽宁省博士基金(20091034)资助。
任永昌(1969—),男,博士后,副教授,主要研究方向为软件成本估算,E-mail:rycryc@sina.com;邢 涛(1978—),男,博士后,副研究员,主要研究方向为智能交通系统工程;刘大成(1968—),男,博士,副教授,主要研究方向为人工智能。

数调整实际数据。

指数平滑法的基本公式是：

$$S_t=\alpha y_t+(1-\alpha)S_{t-1} \tag{1}$$

式中，S_t 为时间 t 的平滑值；y_t 为时间 t 的实际值；S_{t-1} 为时间 $t-1$ 的实际值；α 为平滑常数，取值范围为[0,1]。

由式(1)可知：

(1)S_t 是 y_t 和 S_{t-1} 的加权算数平均数，随着 α 取值的大小变化，决定 y_t 和 S_{t-1} 对 S_t 的影响程度，当 α 取 1 时，$S_t=y_t$；当 α 取 0 时，$S_t=S_{t-1}$。

(2)平滑常数决定于平滑水平以及对预测值与实际结果之间差异的响应速度。平滑常数 α 越接近 1，远期实际值对本期平滑值的下降越迅速；α 越接近 0，远期实际值对本期平滑值影响程度的下降越缓慢。当时间数列相对平稳时，可取较大的 α；当时间数列波动较大时，应取较小的 α，以不忽略远期实际值的影响。

(3)尽管 S_t 包含有全期数据的影响，但实际计算时仅需要 y_t，S_{t-1} 和 α，这就使指数滑动平均具有逐期递推的性质。

(4)根据公式 $S_1=\alpha y_1+(1-\alpha)S_0$，当用指数平滑法时才开始收集数据，则不存在 y_0，也就无法求出 S_1，所以定义 S_1 为初始值。如果找到 y_1 以前的资料，初始值 S_1 就可以确定。数据较少时可用全期平均、移动平均法；数据较多时，可用最小二乘法。如果仅有从 y_1 开始的数据，那么确定初始值的方法有：①取 S_1 等于 y_1；②待积累若干数据后，取 S_1 等于前面若干数据的简单算术平均数。

1.2 公式推导

设 n 个典型项目[3]预估项目的贴近度(即相似程度)为 α_i，$i=1,2,\cdots,n$，将其从大到小排列成一个有序数列为 $\alpha_1,\alpha_2,\cdots,\alpha_n$，相应地典型项目的单方成本为 $E_1,E_2,\cdots,E_n$。就是说，与预估项目最相似的(贴近度最大)典型项目的单方成本为 E_1，次相似的为 $E_2,\cdots$，依此类推，最不相似的为 E_n。

设第 i 个相似项目单方成本的估算值为 E_1^*，估算误差为 $(E_i-E_1^*)$，则第 $i-1$ 个相似项目的单方成本估算值为：

$$E_{i-1}^*=E_1^*+\alpha_i(E_i-E_1^*) \tag{2}$$

式(2)的意思是：对第 i 个相似项目的单方成本估算值进行修正，修正的方法是加上估算误差 $(E_i-E_1^*)$ 和该项目与预估项目的贴近度 α_i 的乘积，然后就把修正后的值作为与欲估项目第 $i-1$ 个相似项目的单方成本的估算值。这样，上式可以改变为：

$$E_{i-1}^*=\alpha_iE_i+(1-\alpha_i)E_1^* \tag{3}$$

依此类推并展开，则可得预估项目的单方成本造价估算值为：

$$\begin{aligned}e^*&=\alpha_1E_1+(1-\alpha_1)E_1^*=\alpha_1E_1+(1-\alpha_1)[\alpha_2E_2+(1-\alpha_2)E_2^*]=\cdots=\alpha_1E_1+\alpha_2(1-\alpha_1)E_2+\alpha_3(1-\alpha_1)(1-\alpha_2)E_3+\cdots+\alpha_n(1-\alpha_1)(1-\alpha_2)\cdots(1-\alpha_{n-1})E_{n-1}+(1-\alpha_1)(1-\alpha_2)\cdots(1-\alpha_n)E_n^*\end{aligned} \tag{4}$$

式中，E_n^* 为估算初始值，取为 n 个典型项目单方成本的算术平均值，即：

$$E_n^*=\frac{1}{n}\sum_{i=1}^{n}E_i \tag{5}$$

预估项目的单方成本估算值就是各相似项目单方成本的加权平均值，这些权值从大到小变化逐渐趋于零，且满足归一化条件，所有权值之和等于 1，即：

$$\alpha_1+\alpha_2(1-\alpha_1)+\alpha_3(1-\alpha_1)(1-\alpha_2)+\cdots+\alpha_n(1-\alpha_1)(1-\alpha_2)\cdots(1-\alpha_{n-1})+(1-\alpha_1)(1-\alpha_2)\cdots(1-\alpha_n)=1 \tag{6}$$

把预测公式改写如下，可以从另一个角度来理解该公式的物理(经济)含义。

$$e^*=E_n^*+\alpha_1(E_1-E_n^*)+\alpha_2(1-\alpha_1)(E_2-E_n^*)+\alpha_n(1-\alpha_1)(1-\alpha_2)\cdots(1-\alpha_{n-1})(E_n-E_n^*) \tag{7}$$

式(7)表明：用各项目单方成本的算术平均值 E_n^* 对预估项目进行估算，精度较低，所以用各相似项目单方成本与算术平均值之差，乘以相应的由贴近度组成的权值来进行调整。相似程度大的项目，权值也大，因而它所起的调整作用越大，相似程度小的项目，权值也小，所起的调整作用也就越小。用相似程度的大小来控制相应项目的调整作用。

在上面按照指数平滑法的思路推导出的预估项目的单方成本估算公式中，贴近度 α_i 就是平滑系数。如果令 $\alpha_1=\alpha_2=\cdots=\alpha_n=\alpha$(大于 0 小于 1 的常数)，则该公式就和一般的指数平滑法的估算公式完全一样。因而，该预测公式不仅具有指数平滑法的全部特点，而且具有某些其它特性。首先，与预估项目越相似(即贴近度越大)的典型项目权值也越大，与预估项目相似程度越小(即贴近度小)的典型项目权值也越小；换句话说，它对相似程度大的典型项目更为重视。其次，采用相似程度(即贴近度)作为平滑系数，由于贴近度的数值一般较大(贴近度越大，表示该典型项目与预估项目越相似，对预估项目的成本估算精度就越高)，因此估算值能够较迅速地反映出成本的变化状态，敏感地跟踪成本的变化趋势，这也是符合要求的。此外，由于权值是呈指数级递减的，衰减非常大，贴近度第 4 大的典型项目其权值已经相当小，通常可以忽略，因此一般只取最相似的 3 个典型项目即可。这就使估算公式大为简化，即：

$$e^*=\alpha_1E_1+\alpha_2E_2(1-\alpha_1)+\alpha_3E_3(1-\alpha_1)(1-\alpha_2)+(1-\alpha_1)(1-\alpha_2)(1-\alpha_3)(E_1+E_2+E_3)/3 \tag{8}$$

由于相似项目与预估项目只是相似不是相同，即存在差异，因此，应该对估算成本进行调整，乘上一个调整系数 λ，则有：

$$e^*=\lambda*[\alpha_1E_1+\alpha_2E_2(1-\alpha_1)+\alpha_3E_3(1-\alpha_1)(1-\alpha_2)+1/3(E_1+E_2+E_3)(1-\alpha_1)(1-\alpha_2)(1-\alpha_3)] \tag{9}$$

式(9)把指数平滑法与模糊数学方法相结合，是进行软件开发成本估算的公式。

1.3 模型建立

已知 n 个典型项目，设 $A_1,A_2,\cdots,A_i$，$i=1,2,\cdots,n$。用 T 表示项目特征集合，此集合元素的确定，以能概括描述项目的特征，并能充分说明问题为原则。常取：$T=$｛开发平台，核心程序，输入输出，开发与应用点数，…｝

记为：$T=\{t_1,t_2,t_3,\cdots,t_j\}\quad j=1,2,3,\cdots,m$

T 的模糊子集合用查德记号记为：

$$T=t_{i1}/t_1+t_{i2}/t_2+\cdots+t_{ij}/t_j \tag{10}$$

式中，t_j 表示项目特征的元素名称；

T_i 表示已知第 i 个项目特征对于集合 T 的模糊子集；

t_{ij} 表示已知项目特征元素对应的隶属函数值(隶属度)；

这样，取预估项目对应的项目特征的模糊子集为：

$$T_0^*=t_1^*/t_1+t_2^*/t_2+\cdots+t_j^*/t_j \tag{11}$$

式中，t_j^* 表示预估项目特征元素所对应的隶属函数值(隶属度)。

根据预测技术中的指数平滑法等有关理论推导出预估项目成本的估算公式：

$$e^* = \lambda * [\alpha_1 E_1 + \alpha_2 E_2 (1-\alpha_1) + \alpha_3 E_3 (1-\alpha_1)(1-\alpha_2) + 1/3(E_1+E_2+E_3)(1-\alpha_1)(1-\alpha_2)(1-\alpha_3)] \quad (12)$$

式中，$\alpha_1, \alpha_2, \alpha_3$ 为预估项目同每个已知项目的贴近度。根据择近原则，贴近度从大到小依次排序，取贴近度大的 3 个已知项目作为估算基础，并满足从大到小的顺序，即：

$$\alpha_1 \geqslant \alpha_2 \geqslant \alpha_3 \quad (13)$$

E_1, E_2, E_3 是 $\alpha_1, \alpha_2, \alpha_3$ 相应的 3 个已知典型项目的某一分项工作的直接成本；λ 为调整系数。考虑选定的 3 个已知典型项目各自对预估项目的影响是不同的，根据对近百个工程调查统计结果，3 项依次各为总值的 60%～65%，25%～30%，3%～7%，即依次乘以系数 1.8，0.8，0.4。因此调整系数 λ 的经验公式为：

$$\lambda = 1 + \frac{1}{m}[1.8 \times (\frac{T_g}{T_{a1}} - 1) + 0.8 \times (\frac{T_g}{T_{a2}} - 1) + 0.4 \times (\frac{T_g}{T_{a3}} - 1)] \quad (14)$$

式中，m 为项目模糊集合中元素个数（项目个数）；T_g 为预估项目的项目模糊关系系数；T_{a1}, T_{a2}, T_{a3} 与 $\alpha_1, \alpha_2, \alpha_3$ 相应的项目中最大的 $\sum t_j$ 为 1，其它各项目的模糊关系系数为与最大值 1 相比所占的比例关系。

2 隶属(度)函数的确定

隶属度，又称隶属函数值或模糊关系系数，是描述事物模糊性的关键。在用模糊理论进行软件成本估算时，要计算预估项目的各项特征与典型项目的各项特征的相似程度，即确定各项特征的隶属度。

确定隶属(度)函数的方法有[4]：概率统计法、模糊统计试验法、专家评定法、历史经验法、分布法等。文中采用二元对比排序法中的相对比较法，现选定 m 个典型项目，通过以下三步来实现：

第一步　建立相对比较级。通过对项目某特征的比较，建立相对比较级如下式：

$$(r_{ij}, r_{ji}) \quad (15)$$

式中，$i=1,2,\cdots,m, j=1,2,\cdots,m, i \neq j$，$r_{ij}$ 表示 i 项目与 j 项目相对照与预估项目的相似程度；r_{ji} 表示 j 项目与 i 项目相对照与预估项目的相似程度。

第二步　建立相及矩阵。志村正道的方法是先建立式(16)和式(17)，然后建立相及矩阵，见式(18)。

$$r'_{ij} = 1, i = j \quad (16)$$

$$r'_{ij} = \frac{r_{ij}}{r_{ij} \vee r_{ji}} \quad i \neq j \quad (17)$$

$$R' = \begin{Bmatrix} 1 & r'_{12} & \cdots & r'_{1m} \\ r'_{21} & 1 & \cdots & r'_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ r'_{m1} & r'_{m2} & \cdots & 1 \end{Bmatrix} \quad (18)$$

第三步　计算 m 个典型项目中某一特征的相似程度的隶属度，见下式：

$$P = (p_{11}, p_{12}, \cdots, p_{1m}) = [\min(1, r'_{12}, \cdots, r'_{1m}), \min(r'_{21}, 1, \cdots, r'_{2m}), \cdots, \min(r'_{m1}, r'_{m2}, \cdots, 1)] \quad (19)$$

3 贴近度的确定

在估算的数学模型中，取贴近度大的 3 个已知项目作为估算基础。“贴近度”的概念是由北京师范大学汪培庄教授提出的[3]。设 A, B 是论域 U 上的两个模糊子集，记：

$$\underline{A} \otimes \underline{B} = \bigvee_{u \in U} (\mu_{\underline{A}}(u) \wedge \mu_{\underline{B}}(u)) \quad (20)$$

$$\underline{A} \oplus \underline{B} = \bigwedge_{u \in U} (\mu_{\underline{A}}(u) \vee \mu_{\underline{B}}(u)) \quad (21)$$

式(20)、式(21)分别叫做 A 和 B 的“内积”和“外积”。符号“$\vee$”表示取最大值，“$\wedge$”表示取最小值，下标 $u \in U$ 表示取最大值和最小值的范围是论域 U 里所有的元素。

设 $\underline{A}, \underline{B}$ 是论域 U 上的两个模糊子集，它们的贴近度定义为下式：

$$(\underline{A}, \underline{B}) = \frac{1}{2}[\underline{A} \otimes \underline{B} + (1 - \underline{A} \oplus \underline{B})] \quad (22)$$

由于内积和外积中取大“$\vee$”和取小“$\wedge$”算子太模糊，丢失的信息太多，只突出隶属度很大或隶属度很小元素的作用，使得贴近度在实际应用中具有很大的局限性。但由于计算特别简单的缘故，尤其适合于手算甚至心算。

4 数据库规模与软件规模的关系

这里的数据库规模(DBscale)，既不是数据库中数据及对象占用的物理空间(文件、设备、表空间)的大小，也不是数据库对象(表、视图、连接、过程)的大小，而是综合指标，它考虑数据库对象的数量和属性、数据库对象占用的物理空间等诸多因素。为了计算简单，在此仅仅考虑“表”一种对象，即只考虑表的数量、结构、元组等因素，而不考虑其它数据库对象。理由是，表是数据库的核心和基础，通常状况下其它数据库对象的数量及复杂性与表的数量及复杂性成正比，通过表估算数据库规模能很好地代表其它数据库对象。

根据表的作用，将表分为字典表、操作表、报表表、系统表等 4 种类型。字典表用于存放基本信息，对数据进行规范化处理、辅助用户输入、保存常规信息等；操作表用于存放业务处理数据，是操作员录入数据库的目的表，是输出信息的源表；报表表是辅助用户输出的表，数据不直接来源于用户操作，而是作为辅助表或中间表；系统表是对应用程序的界面、输入、输出等进行设置，由系统表支持前端程序运行，系统表的运用增加了程序的复杂性。

字典表需要维护程序，字典表增加，维护程序随之增加。字典表用来填写下拉列表框，如果字典表数据过多，则选择或填写下拉列表框速度较慢，用户可输入简码，直接从字典表中查询，要完成这些功能，需要程序来实现，从而增加了软件规模，且字典表数量越多，软件规模也越大。

操作表存放业务数据，需要录入程序。录入程序需要提供的功能有：录入修改界面、检查输入的正确性、连接数据库、保存数据等。每个操作表字段数增加，录入程序也要相应地增加，且操作表字段数量和录入修改程序呈线性关系。操作表数量增加，录入程序也要增加，软件规模就增加。

报表表是辅助打印的中间表，数据需要程序填写。报表表字段数与填写数据程序呈正相关关系。报表表数量增加，填写数据程序也要相应地增加，软件规模就增加。

系统表是对系统进行设置的表。每个系统表的字段数或系统表总数有所增加，维护程序就增加。运用系统表需要应用程序，也使软件规模增加。系统表的运用增加了程序的复杂性，也可以认为是增加了软件规模。

综上所述，数据库规模对软件项目规模有重要影响。数

据库规模大，软件规模就大；数据库规模小，软件规模就小。可以近似认为数据库规模与软件规模呈线性关系。

5 数据库规模估算

要进行精确的数据库规模估算，只有对软件需求非常明确，写出数据要求说明书文档，并在此基础上写出数据库设计说明书文档后才能确定。软件成本估算是软件报价的基础，有时可能在没有充分调研的基础上就需要估算数据库规模，进而估算成本。在不同阶段可以采用不同的方法，估算的准确性也不同。在没有进行详细调研的软件生命周期开始阶段，可采用专家意见法估算数据库规模，在生命周期的中、后期阶段，可根据数据库设计说明书进行较详细的估算。

5.1 影响因素分析

5.1.1 表类型对数据库规模的影响

不同类型的表对数据库规模的影响不同。对于字典表、操作表、报表表，不同的软件开发机构或组织运用的方法基本相同；而系统表运用的方法相差却很大。如果有较高的平台设计人员，可运用系统表来提高效率，减少成本；否则，运用系统表会因增加程序的复杂性而降低效率，增加成本。系统表对软件成本的影响较为复杂，暂不考虑。

根据对专家走访、作者经验及对已完成项目的统计分析，字典表、操作表、报表表对数据库规模的影响系数分别为1.0，5.1，1.4，成本估算时可根据经验进行调整。对于字典表，影响系数 1.0 是基本影响系数，运用 1 次，软件规模增加一个定值，软件成本及数据库规模也相应地增加一个定值，根据经验定值设为 0.1，字典表的运用频率通常为 4～ 10 次，为了计算简单取中间值 7 次，则字典表的影响系数为 1.7。

5.1.2 表结构对数据库规模的影响

表结构对数据库规模的影响主要体现在两个方面[7]，一是属性数量，二是属性数据类型。

(1)属性数量对规模的影响如图 1 所示。

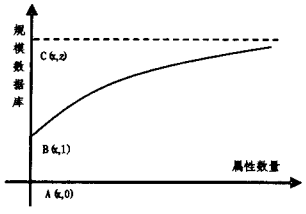

图 1 属性数量对数据库规模的影响

横坐标表示属性数量，纵坐标表示数据库规模，实曲线表示属性数量对数据库规模的影响。属性无论多少，都需要对表进行查询、插入、修改、删除等操作；大部分相同类型的表，属性大体固定在某个范围内，太多或太少的情况较少。横坐标的数量从 x 开始，表明当属性数量小于或等于 x 时，对数据库规模的影响相同，大于 x 时才增加规模。$B(x,1)$表明当属性数量小于或等于 x 时，数据库规模为 1(标准值)。曲线表明当属性数量大于 x 值时，随着属性数量的增加，数据库规模增加，但非正比增加，而是增加速度逐渐放缓。虚线表明当属性数量增加到一定数量时，对数据库规模的影响越来越小，最终数据库规模趋近于定值。x，z 的值大小与表类型有关系，默认值在数学模型中给出。

(2)属性数据类型对规模的影响。对于通常的大型关系数据库管理系统，有二十几种数据类型。以 SQL Server2005 为例，有 25 种数据类型[9]，按操作的难易程度，可分为简单、中等、复杂 3 大类，如表 1 所列。

表 1 数据类型难易程度分类表

难易类别	序号	数据类型	难易类别	序号	数据类型
简单	1	Char	中等	14	Float
	2	Nchar		15	Real
	3	Varchar		16	Money
	4	Nvarchar		17	Smallmoney
	5	Timestamp	复杂	18	Text
	6	Uniqueidentifier		19	Ntext
中等	7	Bit		20	Image
	8	Int		21	Datetime
	9	Bigint		22	Smalldatetime
	10	Smallint		23	Binary
	11	Tinyint		24	Varbinary
	12	Decimal		25	XML
	13	Numeric			

通常数据库规模以 1 为标准，即都是“简单”数据类型，如果“中等”或“复杂”数据类型较多，则要加上调整系数。同一难易程度的字段类型难易程度也不尽相同。如在“中等”类型里，浮点型(12－17)难度大于整型(7－11)；在“复杂”类型里，Image 型难度大于 Datetime 型。为了计算简单，将难度相差不大的归为一类，运用一个难度系数，默认值在数学模型建立时给出。如果需要精确计算，可对每个数据类型赋予一个难度系数。

5.1.3 表元组对数据库规模的影响

表元组数量对数据库规模的影响[7]与属性数量对数据库规模的影响类似，设定固定数量的元组对数据库规模的影响为 1，在此基础上元组增加时，数据库规模增加。如果元组数量较少，不对表进行任何处理就可快速地进行数据操作，不增加数据库规模。如果元组数量较大，就要创建索引，聚集索引要改变表中元组的物理顺序，非聚集索引虽然不改变物理顺序，但要创建指针。索引加快了查询速度，但使更新速度显著降低，因为在更新数据的同时，还要维护索引。创建索引增加了软件开发工作量，增加了数据库规模。如果元组数量特别巨大，还要将索引文件和数据文件保存在不同的磁盘上，以便进行并行的 I/O 操作，使查询性能显著提升。

元组数量对数据库规模的影响分为 3 级：没有影响、较大影响、极大影响，如图 2 所示。

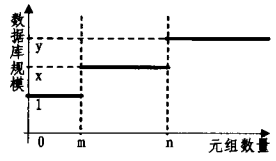

图 2 元组数量对数据库规模的影响

当元组数量介于 0～ m 之间时，没有影响，数据库规模为 1；当元组数量介于 m～ n 之间时，有较大影响，数据库规模 $x>1$；当元组数量大于 n 时，有重大影响，数据库规模 $y>x>1$。根据经验，m，n 的值取 10 万、100 万。x，y 的值取 1.2，1.4。m，n 的取值与 DBMS、服务器 CPU 的数量、类型与主频、服务器内存等有关系，也与表属性的数量及类型有关系。

5.2 数学模型建立

综合考虑表类型、表结构、表元组对数据库规模的影响，建立如下的数据库规模估算公式：

$$DBscale=\sum_{i=1}^{3}(Et\times\sum_{j=1}^{n}(SE\times RE)) \qquad (23)$$

式中，$DBscale$ 为数据库规模。i 取值与公式中其它参数的对应关系如表 2 所列。Et 为表类型对数据库规模的影响系数，默认值见表 2，计算时可根据实际情况改变。

表 2　式(23)中 i 取值与其它参数的对应关系

表类别	i 取值	Et 默认值	n 取值
字典表	1	1.7	字典表的总数
操作表	2	5.1	操作表的总数
报表表	3	1.4	报表表的总数

SE 为表结构对数据库规模的影响，计算公式为：

$$SE=(1+ITEm\times NUMm+ITEc\times NUMc)\times(\log_x y) \tag{24}$$

式中，$(1+ITEm\times NUMm+ITEc\times NUMc)$ 为属性类型对规模的影响，值大于等于 1。$ITEm$ 为 1 个“中等”难易程度数据类型对规模的影响，默认权重值为 0.02。$NUMm$ 为“中等”难易程度数据类型的数量。$ITEc$ 为 1 个“复杂”难易程度数据类型对规模的影响，默认权重值为 0.05。$NUMc$ 为“复杂”难易程度数据类型的数量。

式(24)中，$(\log_x y)$ 为属性数量对规模的影响。x，y 均为大于 0 的正整数，且 $y\geqslant x$，能够保证 $(\log_x y)\geqslant 1$。x 的取值与表类型有关，当前给出字典表、操作表、报表表的默认值分别为 5，10，10。

式(23)中，RE 为表元组数量对数据库规模的影响权重，取值如下：

$RE=$

$$\begin{cases} 1 & \text{当元组数小于 } m \quad m \text{ 默认值 10 万} \\ x \text{ 默认值 1.3} & \text{当元组数介于 } (m,n) \text{ 之间} \quad n \text{ 默认值 100 万} \\ y \text{ 默认值 1.7} & \text{当元组数大于 } n \end{cases} \tag{25}$$

6　成本估算步骤

基于模糊理论的软件开发成本估算步骤流程如图 3 所示。对各估算步骤说明如下：

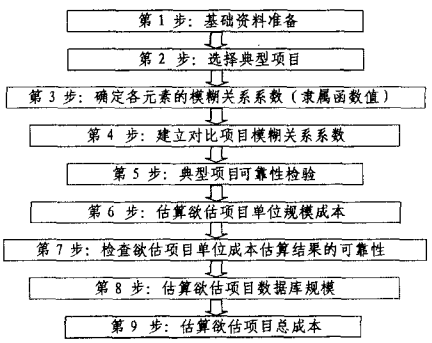

图 3　估算步骤流程图

第 1 步　基础资料准备。选取项目集合中各元素的名称，这些元素的选定要能概括地描述软件有代表性的特征；收集已完工项目的相关资料，建立资料库。

第 2 步　选择典型项目。针对预估项目的具体情况，从资料库中选取与预估项目相似的 5～7 个项目作为典型项目，典型项目的选取要满足以下两个条件：①与预估项目属于同一类型；②项目集合中各元素具有相似性。

第 3 步　确定各元素的模糊关系系数(隶属函数值)。遵循确定隶属(度)函数通常应遵循的原则，可采用二元对比排序法中的相对比较法或其它方法来确定隶属函数值。

第 4 步　建立对比项目模糊关系系数。根据隶属函数值 t_j，算出 $T_i=\sum t_j$，$i=1,2,\cdots,m$，m 最大值为典型项目的数量＋1(预估项目)。并设定 T_j 的最大值为 1，其它各项目的模糊关系系数为与最大值 1 相比所占的比例，在闭区间[0,1]中取值。如工程 F 的模糊关系系数为 $\underline{T_F}=1-\dfrac{T_{\max}-T_F}{T_{\max}}$，其中 $T_{\max}=\max(T_1,T_2,\cdots,T_m)$。

第 5 步　典型项目可靠性检验。步骤为：①列出各典型项目的模糊子集；②根据贴近度确定方法，轮流计算各典型项目的贴近度；③按照择近原则，选取前面 3 个大的贴近度，依次排序使其满足 $\alpha_1\geqslant\alpha_2\geqslant\alpha_3$，以及相应的 3 个典型项目的实际成本 E_1，E_2，E_3；④由调整系数 λ 式(14)，分别计算各典型项目的调整系数 λ 值；⑤利用成本估算式(12)，求出各典型项目的估算成本与实际成本相比较，看是否满足精度要求，精度的计算公式为：

$$\text{精度}=\frac{|\text{估算成本}-\text{实际成本}|}{\text{实际成本}} \tag{26}$$

精度阈值的缺省值设定为 15%，可根据软件特性及精度要求调整，如果软件估算精度较高，则减小阈值。若精度小于阈值，则精度满足要求，则说明典型项目各元素所确定的隶属度可靠；若精度大于阈值，则精度不满足要求，则要对所确定元素的隶属度作适当的局部调整并重新检验精度，直到满足精度要求为止。

第 6 步　估算预估项目单位规模成本。单位规模成本指单位数据库规模成本。对已检验的并满足精度要求的典型项目的相关数据，运用第 5 步的方法，估算预估项目成本。

第 7 步　检查预估项目单位成本估算结果的可靠性。把第 6 步求得的预估项目的单位规模估算成本作为已知，列入典型项目，再次检验各典型项目的精度。若能满足要求，则说明估算结果正确，若不能满足要求，则要对各元素的隶属度作局部调整，按上述方法重算，直至满足精度要求为止。把检验可靠的预估项目作为典型项目，来估算新的预估项目。

第 8 步　估算预估项目数据库规模。按照数据库规模估算的步骤及公式估算数据库规模。

第 9 步　估算预估项目总成本。预估项目总成本是预估项目单位数据库规模成本乘以预估项目数据库规模。

7　模糊理论方法总结

运用模糊理论估算软件开发成本，能从量上把握并处理软件开发成本估算中存在的复杂性，以及各种因素变化的不确定性引起的估算用时较长、估算过程复杂等问题。

(1)软件开发每个阶段的工作方式和工作方法有多种选择，每种选择对成本的影响不同。模糊理论方法引入隶属度的概念，即用某一具体的性质或指标隶属于由该种性质或指标构成集合的程度，来表明该性质或指标对于所在集合的贡献程度，体现了质变与量变之间的辨证规律。

(2)对于“相似”这个模糊概念，可采用模糊数学中的加权海明距离来度量。距离越小说明已完成软件与欲开发软件越相似，可以用该软件的实际成本进行估算；另一方面，加权海明距离中的权重又可以很好地解决软件开发不同过程、不同子系统对软件开发成本的影响。

(下转第 142 页)

knowledge extraction from Web documents[J]. Intelligent Systems,2003,18(1):14-21

[7] Lai Y,Wang R. Towards automatic knowledge acquisition from text based on ontology—centric knowledge representation and acquisition[C]// Proc. of the K—CAP 2003 Workshop on Knowledge Markup and Semantic Annotation (Semannot' 2003). 2003

[8] Schutz A, Buitelaar P. RelExt: A tool for relation extraction from text in ontology extension[C]//Proc. of the 4th Int'l Semantic Web Conf(ISWC). Berlin:Springer,2005:593-606

[9] http://www.fao.org

[10] Christophides V,Karvounarakis G,Plexousakis D,et al. Optimizing taxonomic semantic Web queries using labeling schemes [J]. Journal of Web Sematics,2003,1(2):207-228

[11] Popov B, Kiryakov A, Ognyanoff D, et al. KIM: A Semantic Platform for Information Extaction and Retrieval[J]. Journal of Natural Language Engineering,2004,10(3/4):375-392

[12] Xu J X,Croft W B. Improving the effectiveness of information retrieval with local context analysis[J]. ACM Trans. on Information Systems,2000,18(1):79-112

[13] 刘挺,车万翔,李生. 基于最大熵分类器的语义角色标注[J]. 软件学报,2007,18(3):565-573

[14] Chen Ye-wang,Li Wen,Peng Xin,et al. An Improved Semantic Annotation Method for Documents Based on Ontology[C]// CSWS. 2009

[15] Furnas G W,Landauer T K,Gomez L M,et al. The vocabulary problem in Human-System communication[J]. Communications of the ACM,1987,30(11):964-971

[16] Salton G, McGill M. Introduction to Modern Information Retrieval[M]. New York:McGraw-Hill,1983

[17] Baeza-Yates R, Ribeiro-Neto B. Modern Information Retrieval [M]. New York:Addison-Wesley-Longman,1999

[18] www.flower-sh.cn/article_list.asp? c_id=74&page=4

[19] http://icgr.caas.net.cn/disease/

[20] Gao J, Zhou M, Nie J Y, et al. Resolving query translation ambiguity using a decaying Co-occurrence model and syntactic dependence relations[C]//Järvelin K,Chairs P,Baeza-Yates R,et al.,eds. Proc. of the 25th Annual Int'l ACM SIGIR Conf. on Research and Development in Information Retrieval. Tampere: ACM Press,2002:183-190

[21] Jang M G,Myaeng S H,Park S Y. Using mutual information to resolve query translation ambiguities and query term weighting [C]//Dale R,Church K,eds. Proc. of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics. College Park: Association for Computational Linguistics,1999:223-229

[22] 蔡怡峰,彭鑫,钱乐秋. 面向语义构件检索的交互式查询方案生成[J]. 电子学报,2008,36(8):1631-1636

[23] 陈叶旺. 国家农业本体协同建构与语义检索若干技术研究[D]. 上海:复旦大学,2009

[24] Chang Y,Ounis I,Kim M. Query reformulation using automatically generated query concepts from a document space[J]. Information Processing and Management,2006:42

[25] Zhang M,Song R H,Ma S P. Document refinternet based on semantic query expansion[J]. Chinese Journal of Computers, 2004,27(10):1395-1401

(上接第 134 页)

(3)运用模糊理论方法估算软件开发成本,是用已完成的软件成本估算欲开发的软件成本,可以节省细化软件开发过程或软件组成的时间。有经验的估算人员通过已完成软件的开发成本,利用相似法,加上系数调整,就可以进行估算。受此启发,把相当数量的软件成本及软件特性进行整理输入计算机,并和人工智能技术相结合,就可实现比较准确的成本估算。

(4)模糊理论对软件成本影响因素层次性问题有极好的适用性,能将影响软件开发成本的各因素综合起来,能产生符合客观实际的结果。

软件开发成本估算与模糊理论方法具有很强的结合性,这种结合可以很好地进行软件开发成本估算,同时可以保证结果的客观性,对软件开发起到了积极的指导作用。

参 考 文 献

[1] Li J,Ruhe G,AlEmran A,et al. A Flexible Method for Effort Estimation by Analogy [J]. Empirical Software Engineering, 2006,12(1):65-106

[2] ISBSG. Estimating, benchmarking & research suite release 9 [DB]. Hawthorn, Australia: International software Benchmarking Standards Group—ISBSG,2005

[3] 王祯显,廖小建,杜晓玲. 工程造价快速估算新方法及其应用[M]. 北京:中国建筑工业出版社,1998

[4] 朱训生. 工程管理的模糊分析[M]. 上海:上海交通大学出版社,2004

[5] Boehm BW,Valerdi R,Lane J,et al. COCOMO suite methodology and evolution[J]. CrossTalk:The Journal ofDefense Software Engineering,2005,18(4):20-25

[6] Briand L C,Wieczorek I. Resource Estimation in Software Engineering[M]//Marcinak J J,ed. Encyclopedia of Software Engineering. New York:John Wiley & Sons,2002:1160-1196

[7] 任永昌,邢涛,于忠党,等. 数据库规模估算数学模型研究[J]. 微电子学与计算机,2009,26(7):36-39

[8] 李明树,何梅,杨达,等. 软件成本估算方法及应用[J]. 软件学报,2007,18(4):775-795

[9] Dewson R. Beginning SQL Server 2005 for Developers[M]. Beijing:Post & telecom press,2006

[10] Nguyen H T,Wang T H,Wu B L. On probabilistic methods in fuzzy theory[J]. International Journal of Intelligent Systems, 2004,19(1):78-80

[11] Ruhe M,Jeffery R,Wieczorek I. Cost estimation for web applications[C]//Proc. 25th Int'l Conf. Software Engineering. Los Alamitos,CA:IEEE Computer society Press,2003

[12] Jorgensen M. A review of studies on expert estimation of software development effort[J]. Journal of Systems and Software, 2004,70(1/2):123-128