

基于 EEP 法的一维有限元自适应求解^{*}

袁 驹, 和雪峰

(清华大学 土木工程系, 北京 100084)

(龙驭球推荐)

摘要: 基于新近提出的一维有限元后处理超收敛算法——单元能量投影(EEP)法, 将有限元自适应求解问题转化为对超收敛解答的自适应分段多项式插值问题; 对于大多数问题, 一步便可获得满意的有限元网格划分, 在该网格上再次进行有限元计算, 一般即可获得满足用户给定的误差限的有限元解答。即便未能完全满足精度要求, 一般只需局部细分加密网格一至二步即可。该法简单实用、高效可靠, 是一个颇具优势和潜力的自适应方法。以二阶椭圆型常微分方程模型问题为例, 对该法的基本思想、实施策略及具体算法做一介绍, 并给出有代表性的数值算例用以展示该法的优良性能和效果。

关键词: 有限元法; 自适应求解; 超收敛; 单元能量投影法; 常微分方程

中图分类号: O242.21 **文献标识码:** A

引 言

自适应求解功能是成功的数值计算方法争相追求的现代目标, 特别是工程当中应用最为广泛的有限元法(FEM), 其高效可靠的自适应求解算法更是近年来研究的热点与难点。有限元的自适应求解和传统求解从性质上讲有很大的不同, 在相同的定解条件下

- 传统的求解方法: 输入网格, 得到该网格上的解答, 解答质量和精度由用户判断。
- 自适应求解方法: 输入对解答精度要求的误差限, 得到满足误差限的解答, 而求解网格由算法自适应地调整, 解答质量和精度有可靠保证。

为了成功地实施自适应求解, 必须有可靠的误差估计手段和灵活高效的网格划分和调整技术。在有限元误差估计方面, 目前主要有两大类方法。其一是以 Babuska^[1]等为代表所提出的基于平衡方程残差计算的误差估计方法, 这类估计通常具有相对完备的数学分析基础, 但所给出的误差形式相对复杂, 有时还需要作二次整体计算。另一类则是以 Zienkiewicz 和 Zhu 提出的分片磨光法^[2,3](superconvergent patch recovery, 简称 SPR)、林群^[4]等所开创的有限元插值后处理方法、陈传森^[5]等所建立的单元正交分析法以及其他同类方法为代表的基于后处理超收敛计算的误差估计方式, 这类方法通常具有简便易行的特点。在网格自适应调整方面, 多数

* 收稿日期: 2005_10_09; 修订日期: 2006_08_10

基金项目: 国家自然科学基金资助项目(50278046)

作者简介: 袁驹(1953—), 男, 北京人, 教授, 博士(联系人, Tel: + 86_10_62786185; Fax: + 86_10_62771132; E_mail: yuans@tsinghua.edu.cn);
和雪峰(1975—), 男, 云南人, 博士(E_mail: hxf01@mails.tsinghua.edu.cn)。

方法都是采用能量模来度量误差的大小, 根据各种类型的局部误差指标调整网格; 调整方式则可分为 h 型(网格细分加密法)、 p 型(提升单元阶次法)、以及 hp 型(混合方法)等。

作为非有限元法的自适应分析方法, 值得一提的是国际上著名的常微分方程求解器 COLSYS^[6,7], 它采用样条配点法求解一般的常微分方程(组)边值问题。在自适应方面, COLSYS 通过粗细两套网格(细网格是粗网格的简单二分加密)上解的对比获得误差估计, 由此误差估计确定进一步网格调整的策略。由于 COLSYS 在常微分方程自适应求解中有着出色的表现, 在本文的研究中, 将它列为算法性能参考和追比的对象之一。

本文第一作者及其课题组经过多年的探索和研究, 从结构力学中矩阵位移法的实施中受到启发, 以有限元法数学理论中的投影定理为理论依据, 对一维有限元提出了一种新型的、自然而合理的后处理超收敛计算方法, 称为 EEP(element energy projection)法^[8,9]。该法简便易行、行之有效、效果显著, 可以快速高效地给出任一单元上任一点处的高质量的超收敛解答(包括位移和应力)。这里, 高质量至少有两方面的含义: 一是数值逼近精度高, 如具有逐点一致的逼近精度; 二是形状逼近保真, 如具有足够的连续性、光滑性和凹凸性。简单地讲, 既“保值”, 又“保形”。

EEP 法获得的高质量超收敛解, 为有限元的进一步自适应求解提供了重要的基础, 也提供了新的发展空间。经过一系列的探索、研究和尝试, 对于一维有限元问题, 找到了一套能够充分利用 EEP 超收敛算法之优良特性的自适应求解方案。该方法将有限元自适应求解问题转化为对超收敛解答的自适应分段多项式插值问题, 对于大多数问题, 一步便可获得满意的有限元网格划分, 在该网格上再次进行有限元计算, 一般即可获得满足用户给定的误差限的有限元解答。该法简单实用、快速高效, 是一个颇具优势和潜力的自适应方法。文中以椭圆型二阶常微分方程模型问题为例, 对该法的基本思想、实施策略和具体算法做了介绍, 并给出有代表性的数值算例用以展示该法的优良性能和效果。

1 模型问题及 EEP 超收敛解

1.1 模型问题和有限元解

本文考虑如下的二阶常微分方程两点边值问题

$$\begin{cases} Lu \equiv -(p u')' + qu = f, & 0 < x < 1, \\ u(0) = 0, & u'(1) = 0, \end{cases} \quad (1a)$$

$$(1b)$$

式中 $p > 0$, $q \geq 0$, $(\cdot)' = d(\cdot)/dx$, L 为式中定义的椭圆型自伴线性微分算子。对应的物理模型如图 1 所示, 是变截面杆件轴向弹性变形问题, 其中 u 为轴向位移、 f 为轴向分布荷载、 q 为分布弹簧的刚度, 杆件截面的拉压刚度为 p , 杆长为单位值。

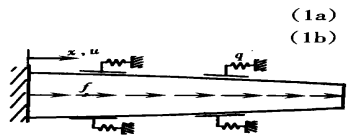


图 1 二阶问题物理模型

用有限元分析上述问题, 得到有限元解 u^h 。熟知, 常规有限元解 u^h 在单元端点具有超收敛性质, 即对于足够光滑的解答, m 次单元蹬匪结点位移可达到 h^{2m} 阶的收敛精度^[10,11]。然而, 对于常规的有限元来讲, 内力是通过位移形函数求得得到的, 其精度呈数量级地下降, 特别是端结点处的内力, 往往是精度最差的点, 而且通常单元内任意一点的位移也是由端点位移插值得到的, 精度也远达不到与结点位移同样的精度。

1.2 超收敛计算公式

文献[9]提出了一种有限元后处理的超收敛计算方法,即 EEP(单元能量投影)法。用该法可以给出任一点的位移和应力(导数)的超收敛解。大量研究和数值计算表明,EEP 超收敛解与结点位移具有相同的超收敛阶。这里,直接给出该法的计算公式,即对于 m 次单元,任一单元内任意一点 $x_a \in (x_1, x_2)$ 处的超收敛位移 u_a^* 和超收敛导数 $u_a^{*'} (应力)$ 由下式给出^[9]

$$u_a^* = u_a^h + \frac{h}{p_a} \left[N_{1a} \int_{x_1}^{x_a} (f - Lu^h) N_2 dx + N_{2a} \int_{x_a}^{x_2} (f - Lu^h) N_1 dx \right], \quad (2a)$$

$$u_a^{*'} = u_a^{h'} + \frac{h}{p_a} \left[N'_{1a} \int_{x_1}^{x_a} (f - Lu^h) N_2 dx + N'_{2a} \int_{x_a}^{x_2} (f - Lu^h) N_1 dx \right], \quad (2b)$$

其中 x_1 和 x_2 为单元端点坐标, $(\)_a$ 表示取在 x_a 点的值, u^h 和 $u^{h'}$ 为常规 m 次有限元解的位移及其导数, N_1 和 N_2 为线性单元的两个形函数, $h = x_2 - x_1$ 。

为讨论方便,以下总记 u^h 为常规有限元解, u^* 为式(2a)计算的超收敛解, u 为原问题(1)的精确解。

1.3 超收敛解答特性

式(2)给出的超收敛解答具有一些颇合人意的优良特性,摘述如下

- 1) 任一点的位移和导数与结点位移具有相同的收敛阶,且精度大致相当。
- 2) 公共结点两侧单元各自算出的应力自动连续,无需人为磨光。
- 3) 超收敛位移在单元端点上自动地等于端点有限元解。
- 4) 自由端点的应力自动为精确值,即可精确满足自然边界条件。
- 5) 任意一点的位移和导数都是用能量积分形式计算出来的,避免直接微分导致精度丧失。
- 6) 超收敛计算属于后处理计算,独立于常规有限元的分析求解。
- 7) 超收敛公式计算可以在各个单元上独立完成。
- 8) 当两端边界条件都是零位移时,即使整个区域只取一个线性单元,公式(2)仍然给出相当不错的超收敛解答,这是常规有限元(只能得到零解)做不到的。

1.4 超收敛解答示例

以如下的 Zienkiewicz 问题为例

$$\begin{cases} Lu \equiv -u'' + u = f, & 0 < x < 1, \\ u(0) = 0, u(1) = 0, \end{cases} \quad (3)$$

其中 $u = x^2 - \sinh 4x / \sinh 4$, f 由方程导出。此例题最初由 Zienkiewicz 和 Zhu^[2,3] 等用来进行切片磨光法(SPR)的研究,故称其为 Zienkiewicz 问题。这里,对整体区域用一个线性元求解,得 $u^h (\equiv 0)$, 并在此 u^h 的基础上用式(2a)计算 EEP 超收敛解 u^* 。图2给出了 u^h 、 u^* 和 u 的曲线比较,其中 u^h 处处为 0。实际上,一个线性元求解是失效的;尽管如此,超收敛解 u^* 与精确解 u 相比无论从数值上还是形状上都十分接近,对常规有限元解的大幅度改进作用显而易见。

2 新型自适应求解策略

2.1 自适应目标

一个优秀的自适应求解算法应该尽可能具备以下几个特征

- 1) 解答一致收敛: 能够逐点估计误差、控制误差; 如采用最大模度量误差并控制误差。
- 2) 误差均匀分布: 各个单元的最大误差大致相等, 减少冗余精度。
- 3) 非均匀网格: 自动生成适应解答变化的非均匀网格划分。
- 4) 最少单元数: 尽可能使用最少的单元, 减少冗余计算。
- 5) 最少迭代步: 尽可能少的自适应迭代计算步骤, 最大限度地提高自适应求解效率。

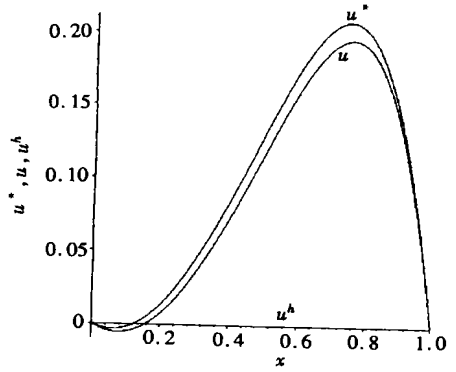


图2 Zienkiewicz 问题解答的比较

以上5个目标中, 前4个是对解答质量的要求, 而第5个是对求解过程之效率的要求。应该说, 上述5个目标要想全面、完美的实现, 一般来讲是很困难的, 其中最大的难点莫过于真解事先未知。

本文的自适应求解目标为: 事先给定误差限 T , 寻求一个优化的有限元网格 π^* , 使得该网格上的有限元解答 u^h , 按最大模度量满足 $\|u - u^h\|_{\max} \leq T$ 。

2.2 网格生成思路

有限元数学理论中有一个为人熟知的重要理论, 即采用能量模来度量, 在同一个网格上, 有^[10]

$$\|u - u^h\| \leq \|u - u_1\|, \quad (4)$$

其中, u_1 是在 u^h 的网格上对 u 的同阶次插值解。这个公式, 将真解 u 、插值解 u_1 和有限元解 u^h 联系在一起。实际上, 有限元的误差估计很多都来自于这一公式。式(4)意味着: 在相同网格上, 若插值解满足误差限 T , 则有限元解亦然; 亦即, 若找到一个 $\|u - u_1\| = T$ 的网格 π_1 , 则也就找到了 $\|u - u^h\| \leq T$ 的网格。但是, 由于 u 事先未知, 因此难以由 $\|u - u_1\| = T$ 构造 π_1 。然而, 对于自适应求解, 问题归结为如何求得网格 π_1 (或者对 π_1 的一个很好的近似), 而并不直接需要真解 u 。考虑到 u^* 无论是数值上还是形状上都更接近于 u , 因此用 u^* 替代 u 来对 u^* 做插值 u_1^* 使得 $\|u^* - u_1^*\| = T$, 所得到的网格 π_1^* 将是对 π_1 很好的近似。这便是本文网格生成的基本思路。

以上的分析叙述采用整体结构的能量模估计误差 $\|u^* - u_1^*\|$, 使用不便。从实用上看, 能够逐单元用最大模估计和控制误差最为方便; 同时由于 EEP 超收敛解的点点超收敛特性, 使用最大模也成为合理而自然的选择。下面考虑这一情况。

2.3 单元误差估计

由有限元及其超收敛解的误差估计理论知, 位移 u^h 在单元端点上具有 h^{2m} 阶超收敛性质, 而在单元内部与插值具有相同的 h^{m+1} 收敛阶; 这就意味着, 随着网格加密, 最大误差 $\max |e^h|$ 发生在单元内部, 而且比端点位移的误差 e_i^h 要大得多。有限元中还有一个超收敛估计, 即

$$\int_0^1 (u - u^h) dx = O(h^{m+2}),$$

这意味着, 误差在整个区域上急剧变号^[10], 而且如文献[10]所述, 误差在单元内也变号。亦即, 单元 e 上最大误差为最大正、负误差中的较大者

$$\max_e |e^h| = \max(\max_e(e^h), \max_e(-e^h)) \quad (\text{在单元 } e \text{ 上}) \quad (5)$$

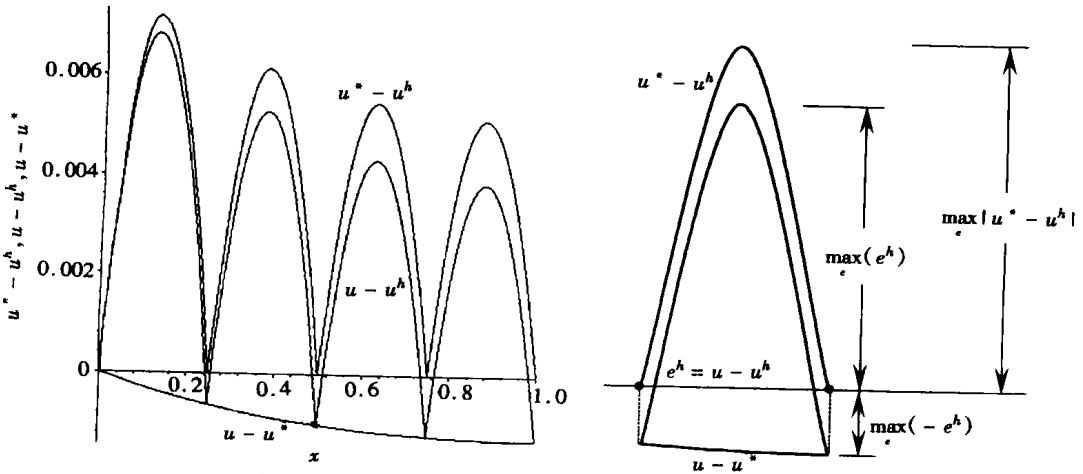
为便于说明,图3给出了一个典型问题的误差分布示例,其中 π_1^* 是4个线性元的均匀网格,在其上求解 u^h ,进而得到 u^* 。从图中3个误差 $e^h = u - u^h$, $e^h = u^* - u^h$ 和 $e^h = u - u^*$ 的曲线比较,可以看出: u^h 是网格 π_1^* 上对 u^* 的插值,因此 $u^* - u^h$ 在单元两端点为0,在单元内部达最大值;又如图3(b)所示, $u^* - u^h$ 的最大值与单元内有限元解 u^h 的最大正、负误差之和大致相等(也为大量的数值试验所验证),即

$$\max_e |u^* - u^h| \approx \max_e(e^h) + \max_e(-e^h) \quad (\text{在单元 } e \text{ 上}) \quad (6)$$

这样,结合式(5)和式(6),有如下的近似估计式

$$\max_e |e^h| \leq \max_e |u^* - u^h| \quad (\text{在单元 } e \text{ 上}), \quad (7)$$

这是一个简便易行的误差估计关系式。对于高次单元,式(6)并不适用,但并不影响式(7)的结论;篇幅所限,不展开讨论。大量数值试验表明,这一估计对于椭圆形问题是可靠的,而且随着网格加密而越来越准确。



(a) 网络上误差比较

(b) 图为(a)中单元误差比较示意图

图3 π_1^* 上误差分布比较

2.4 自适应有限元求解

由式(7),自适应有限元求解便可归结为求得一个最佳网格 π_1^* 使得在其上求得的 u^h 和 u^* 对任意单元 e 都满足

$$\max_e |u^* - u^h| \leq T, \quad (8)$$

而最佳网格 π_1^* 的生成,便可以通过对 u^* 实施自适应插值,使其插值 u_1^* 在各单元上满足下式来实现

$$\max_e |u^* - u_1^*| \leq T, \quad (9)$$

这样,最佳网格生成就转化为对 u^* 的自适应插值问题。

如此,就得到了一个自适应有限元求解的基本实施方案

- 1) 对整体区域取一个初始网格 π_0 (通常一个单元即可)。
- 2) 在 π_0 上求有限元解 u^h ,并用EEP法式(2)计算各单元的超收敛解 u^* 。
- 3) 对 u^* 进行自适应插值逼近,使其满足式(9),得到网格 π_1^* 。

- 4) 在 π_k^* 上求解新的有限元解 u^h 和超收敛解 u^* 。
- 5) 在 π_k^* 上逐单元校验 u^h 和 u^* 的误差, 若均满足式(8), 则终止自适应求解过程。
- 6) 若不满足, 转到步骤 3 构建新的网格。

以上, 每实施一次步骤 3~ 步骤 4 完成一步自适应计算。大量计算表明, 绝大多数情况下, 一步自适应计算即可获得满意的结果, 第 6 步通常是不必要的。

3 自适应插值

上述自适应求解方案将问题归结为对超收敛解的分段多项式自适应插值问题。对此, 本文提出两种算法, 一种是以最少单元数为目标, 称为精准法; 另一种是以高效率为目标, 称为消差法。

3.1 精准法

本法对求解区域从左至右逐个确定自适应插值的单元长度, 在每个单元上使式(9)中的等号成立。做法可以有多种, 本文采用一种比较简单直观离散点搜索和二分找界的方法。设求解区域为 $[0, 1]$, 单元长度的允许误差限为 T_h , 下面以第一个单元长度 h_1 的确定为例, 简单地加以介绍。

- 1) 令 $i = 0$, 给定单元 1 的初始长度 $h_1^{(0)} = 1/2$ 并以两端点为上下界: $h_{1l} = 0, h_{1u} = 1$ 。
- 2) 构造当前单元上的插值函数 u_1^* (采用和有限元相同次数的多项式插值)。
- 3) 将当前单元等分为 N 个子区间, 在 $N - 1$ 个内部分点上计算插值误差 $u^* - u_1^*$ (两端点误差为 0), 并找出误差最大值 $e_1^* = \max |u^* - u_1^*|$ 。
- 4) 若 $e_1^* > T$, 则找到更小的上界, 令 $h_{1u} = h_1^{(i)}$, 再设 $h_1^{(i+1)} = (h_{1l} + h_{1u})/2$ 。
- 5) 若 $e_1^* < T$, 则找到更大的下界, 令 $h_{1l} = h_1^{(i)}$, 再设 $h_1^{(i+1)} = (h_{1l} + h_{1u})/2$ 。
- 6) 若 $e_1^* = T$ 或 $h_{1u} - h_{1l} \leq T_h$, 则停止, 并取 $h_1 = (h_{1l} + h_{1u})/2$; 否则, 令 $i = i + 1$ 并转到步骤 2)。

这一算法的优点是简单直观, 而且仅计算函数值(无需导数值), 且 N 越大, 精度越高。精准法是一个比较耗时的方法, 主要用于理论研究和验证。

3.2 消差法

本法通过不断地在最大误差处插入新的插值结点, 从而有效地消去最大误差, 再在新的网格上构造插值, 直至式(9)在整个区域所有插值单元内都得到满足。设求解区域为 $[0, 1]$, 以下是这一方法实施的步骤。

- 1) 将整个求解区域取为一个单元。
 - 2) 构造当前单元上的插值函数 u_1^* (采用和有限元相同次数的多项式)。
 - 3) 将当前单元等分为 M 个子区间, 在 $M - 1$ 个内部分点上计算插值误差, 并找出误差最大值 $e_1^* = \max_j |u^* - u_1^*|_j (j = 1, 2, \dots, M - 1)$ 和相应的坐标位置。
 - 4) 若 $e_1^* < T$, 则标记该单元已经通过, 不作任何处理。
 - 5) 若 $e_1^* \geq T$, 则在 e_1^* 处插入一个插值结点, 将原单元分为两个子单元。
 - 6) 对每个新生成的子单元, 重复做 2) 至 5) 步。
 - 7) 递归地做 2) 至 6) 步, 直至所有生成的子单元上都有 $e_1^* < T$ 为止, 由此得到 π_k^* 。
- 消差法是一个简单实用且高效可靠的方法。每计算 $M - 1$ 个点的插值误差值(即计算 M

- 1 个超收敛解), 就可以增加一个单元(插入了一个结点) 或者验证通过了一个单元, 而且随着单元数的增多, $M - 1$ 个计算点也越来越密集, 因此也越来越准确可靠。实际上, 设最后得到的网格共有 N_e 个单元, 则不难估计计算超收敛解的次数为 $(2N_e - 1) \cdot (M - 1)$ 。消差法生成的网格在单元数上虽然会比精准法多, 但一般不超过两倍, 而由其生成的网格具有更好的安全储备。

3.3 完整算法

本小节基于最大模和消差法, 给出自适应有限元求解的完整而具体的算法。

1) 对整体区域取单元数为 N_e 的初始网格 π_0 (通常 $N_e = 1$ 即可), 令 $\pi = \pi_0$ 。

I 有限元求解部分

2) 在当前网格 π 上用 m 次元求有限元解 u^h 。

II 网格细分加密

3) 逐单元循环: ie 由 1 到 N_e 循环, 实施以下 4) ~ 8) 的步骤。

4) 将当前单元等分为 M 个子区间, 用式(2) 计算 $M - 1$ 个内部分点的超收敛解 u_j^* ($j = 1, 2, \dots, M - 1$)。

5) 在当前单元上建立超收敛解的 m 次多项式插值, 计算 $M - 1$ 个内部分点的误差, 并找出误差最大值 $e_1^* = \max_j |u^* - u_1^*|_j$ ($j = 1, 2, \dots, M - 1$) 及其相应位置。

6) 若 $e_1^* < T$, 标记该单元已通过, 回到步 3) 处理下一个单元。

7) 若 $e_1^* \geq T$, 则在 e_1^* 处插入一个结点, 将当前单元分成两个子单元, 继续下述步骤。

递归循环迭代: 对每个新生成的子单元作步 3 到 7 (此时 $N_e > 1$), 若子单元又细分, 则重复这一递归循环迭代, 直到所有新生成的单元都无需再细分。

8) 回到步 3) 处理 π 的下一个单元。

9) 所有 π 的单元处理完毕, 得到新的网格 π_1^* 和新的单元数 N_e^* (记为一次自适应插值完成)。

10) 若 $N_e^* > N_e$, 重置 $N_e = N_e^*$ 和 $\pi = \pi_1^*$, 返回步 2。

11) 若 $N_e^* = N_e$, 结束自适应求解。

以上自适应算法中网格生成的特点是一步基本到位(即一步就从一个单元生成为多个单元的不规则网格), 后面的自适应步骤只是局部细分加密, 整个自适应迭代计算步骤非常节省, 一般一至二步即可。这一点应该归功于 EEP 法计算出来的高质量的超收敛解。

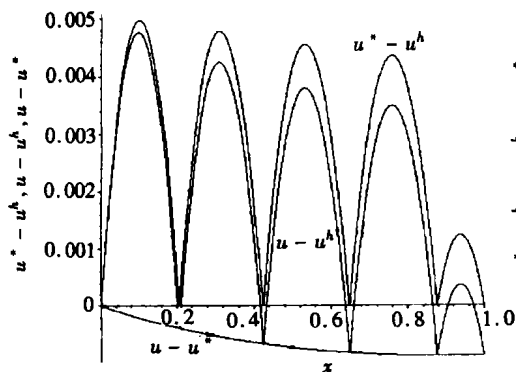
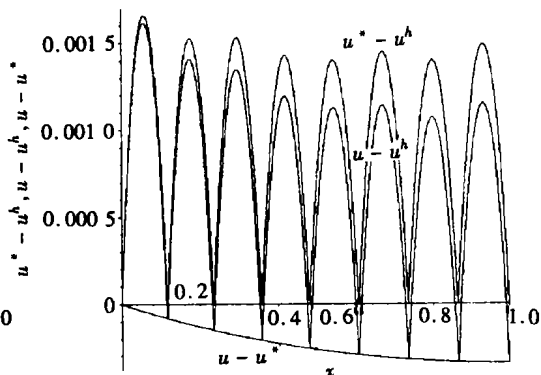
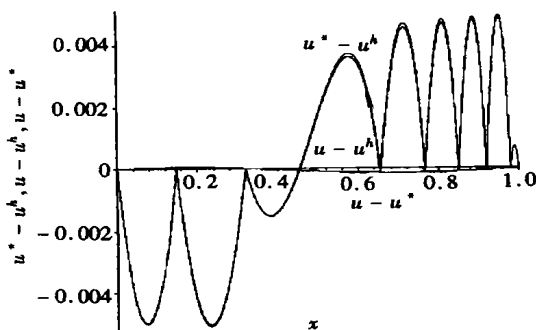
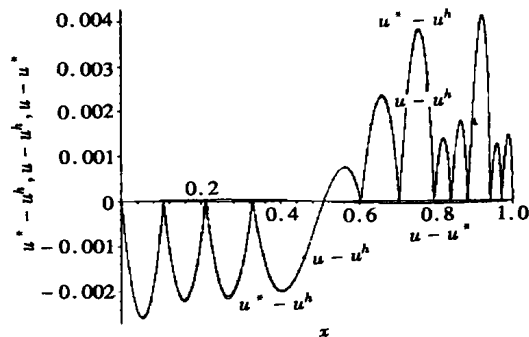
3.4 超收敛解的计算

超收敛解 u^* 的精确计算是一个基本问题, 其中的关键又在于式(2) 中的积分如何实现。

对于一些精度要求不高的工程问题, 一般的 Gauss 数值积分就足够了。但是对于一些高精度要求的研究性问题或刁难问题, 尤其当荷载具有奇异性或振荡性时, 一般的数值积分便难以保证积分结果的质量。此时, 宜采用自适应数值积分以确保积分值具有足够的精度, 比如至少和用户提供的误差限 T 一致。由于本文的理论研究不希望引进数值积分的误差, 因此采用了国际上著名数值计算软件包 QUADPACK 中的自适应求积程序。该程序采用 15 至 61 个 Gauss-Kronrod 点自适应求积, 对奇异和振荡等各类特殊性问题具有良好的适用性, 且其 FORTRAN 代码是公开的。实际计算表明, 此代码所提供的自适应求积算法能够很好地满足本文算法对积分质量的要求。

4 数值算例

本节给出若干有代表性的数值算例,包括奇异问题和高振荡问题。以下算例中,统一采用最大模对绝对误差进行控制,误差限设为 $T = 0.005$ 。初始网格统一地取为 1 个单元。例 1~ 例 3 用线性单元求解,分别采用精准法和消差法进行自适应插值;其中例 2~ 例 3 的初始有限元的解退化为 0 解。例 4~ 例 6 分别用线性元至 4 次元求解,采用更为实用的消差法进行自适应插值;其中用线性元求解时,例 5 的初始有限元解退化为 0 解。又统一地记: π_1 为对 u 进行线性自适应插值求得的网格, π_1^* 为对初始网格上计算出的 u^* 进行线性自适应插值求得的网格。数值积分、精准法和消差法的数值计算中的精度控制都远远超过给定的误差限 T , 以排除他们可能引入的误差。

图 4 精准法 π_1^* 上解的误差分布图图 5 消差法 π_1^* 上解的误差分布图图 6 精准法 π_1^* 上解的误差分布图图 7 消差法 π_1^* 上解的误差分布图表 1 精准法 π_1 和 π_1^* 的结点分布比较

	x_0	x_1	x_2	x_3	x_4	x_5
π_1	0.000	0.208	0.431	0.667	0.913	1.000
π_1^*	0.000	0.204	0.417	0.640	0.876	1.000

例 1 定解条件为 $Lu = -u'' + u = 1$, $u(0) = 0$, $u'(1) = 0$, 其精确解是 $u = -(e^x + e^{-2x})/(1 + e^2) + 1$ 。精准法求得 5 个单元, π_1 和 π_1^* 的结点分布比较见表 1, π_1^* 上解的误差分布如图 4。消差法求得 8 个单元, π_1 和 π_1^* 的结点分布比较见表 2, π_1^* 上解的误差分布如图 5。可见精准法 π_1^* 与 π_1 相当一致, 而消差法则几乎完全一致。2 种方法均全面严格满足了误差

限, 消差法的结果更为安全可靠。

例 2 前述的 Zienkiewicz 问题。精准法求得 8 个单元, π_1 和 π_1^* 的结点分布比较见表 3, π_1^* 上解的误差分布如图 6。消差法 π_1 求得 10 个单元; π_1^* 求得 11 个单元, 较 π_1 更为密集; 结点分布比较见表 4, π_1^* 上解的误差分布如图 7。二者的解答均全面严格满足误差限。

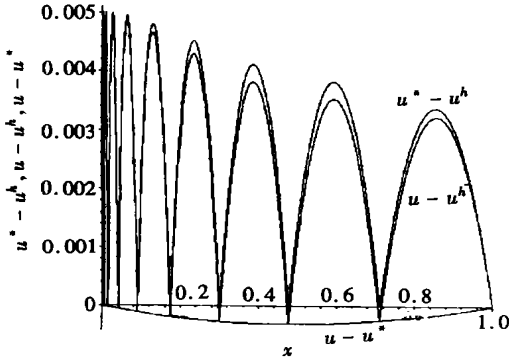


图 8 精准法 π_1^* 上解的误差分布图

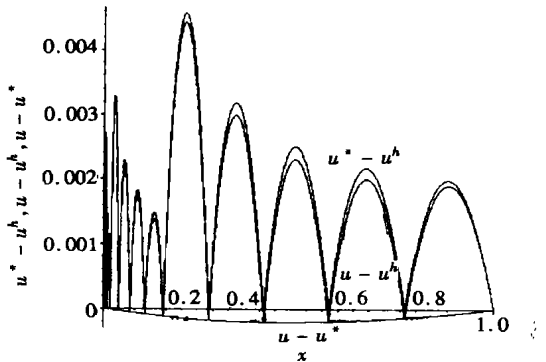


图 9 消差法 π_1^* 上解的误差分布图

表 2 消差法 π_1 和 π_1^* 的结点分布比较

	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
π_1	0.000	0.115	0.235	0.355	0.480	0.607	0.735	0.867	1.000
π_1^*	0.000	0.118	0.235	0.358	0.480	0.605	0.735	0.865	1.000

表 3 精准法 π_1 和 π_1^* 的结点分布比较

	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
π_1	0.000	0.148	0.319	0.668	0.778	0.861	0.930	0.988	1.000
π_1^*	0.000	0.148	0.320	0.655	0.766	0.850	0.920	0.980	1.000

表 4 消差法 π_1 和 π_1^* 的结点分布比较

	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}
π_1	0.000	0.120	0.240	0.358	0.632	0.716	0.800	0.850	0.900	0.950	1.000	
π_1^*	0.000	0.105	0.210	0.328	0.602	0.700	0.790	0.835	0.880	0.940	0.970	1.000

表 5 精准法 π_1 和 π_1^* 的结点分布比较

	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
π_1	0.000 0	0.000 4	0.003 6	0.014 4	0.040 0	0.090 0	0.176 4	0.313 6	0.518 4	0.810 0	1.000 0
π_1^*	0.000 0	0.000 4	0.003 6	0.014 4	0.040 0	0.089 6	0.173 6	0.300 9	0.477 4	0.711 7	1.000 0

表 6 消差法 π_1 和 π_1^* 的结点分布比较

	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	...	x_{14}
π_1	0.000 0	0.000 2	0.000 7	0.002 4	0.008 1	0.015 7	0.027 0	0.052 2	0.090 0	...	1.000 0
π_1^*	0.000 0	0.000 2	0.000 7	0.002 4	0.008 1	0.015 7	0.027 0	0.052 2	0.090 0	...	1.000 0

例 3 控制方程及边界条件同式(3), 令 $u = \sqrt{x} - x, f$ 由微分方程导出。这是一个典型的奇异问题, 导数在 $x = 0$ 处奇异。精准法求得 10 个单元, π_1 和 π_1^* 的结点分布比较见表

5, π_1^* 上解的误差分布如图 8。消差法求得 14 个单元, π_1 和 π_1^* 完全一致, 结点分布比较见表 6, π_1^* 上解的误差分布如图 9。二者的解答均全面严格满足误差限。

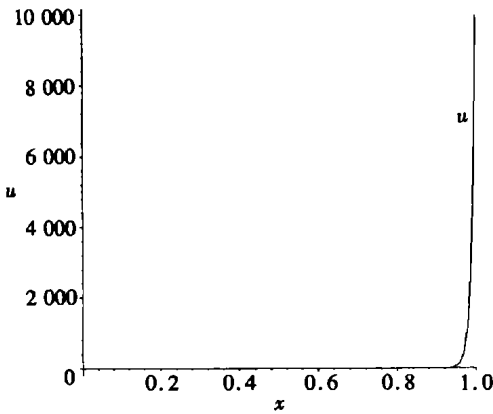


图 10 u 的分布示意图

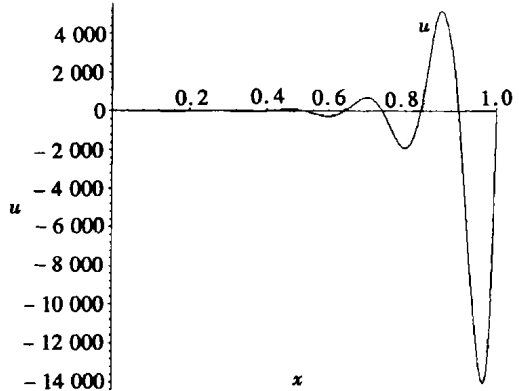


图 11 u 的分布示意图

值得一提的是, 本例如果用 COLSYS 求解, 配点数采用程序的默认配置^[7], 只对解函数控制误差(导数奇异, 无法控制误差), 通过其自适应求解最终并不能求得全面满足误差限的解答。如果要改善其结果, 必须人为提供相当密集的初始网格或非均匀网格。

表 7 各阶次单元求解的结果

	自适应计算次数	最终网格单元数	$\max_e u - u^h $	h_{\max}	h_{\min}
线性元	1	1 961	0.004 992	0.011 6	1.23E- 5
2 次元	1	182	0.004 919	0.085 9	1.09E- 4
3 次元	1	69	0.004 490	0.164 8	2.64E- 4
4 次元	1	57	0.002 966	0.233 4	6.24E- 4

表 8 各阶次单元求解的结果

	自适应计算次数	最终网格单元数	$\max_e u - u^h $	h_{\max}	h_{\min}
线性元	2	528 4	0.004 999 8	0.017 2	3.63E- 5
2 次元	2	486	0.004 960 1	0.028 8	2.55E- 4
3 次元	1	166	0.004 831 3	0.052 2	6.25E- 4
4 次元	2	111	0.003 485 6	0.082 6	1.35E- 3

表 9 各阶次单元求解的结果

	自适应计算次数	最终网格单元数	$\max_e u - u^h $	h_{\max}	h_{\min}
线性元	1	956	0.004 973	0.011 5	3.25E- 5
2 次元	1	351	0.004 663	0.031 8	7.18E- 4
3 次元	1	245	0.002 955	0.042 7	8.73E- 4
4 次元	1	236	0.003 830	0.051 8	8.31E- 4

例 4 控制方程同式(3), 令 $u = 10\,000x^{100}$, f 和两端 u 的边界条件由 u 导出(即非齐次边界条件)。 u 的分布如图 10 所示。只采用消差法求解。表 7 是线性元、2 次元、3 次元、4 次元求解的结果。可见, 预设的误差限得到了全面严格的满足; 最终的网格结点分布自动地左疏右密, 以适应右端的剧烈变化。对于求解器 COLSYS 来讲, 本例的表现与例 3 相同, 即采用常规

解法的自适应求解亦不能获得全面满足误差限的解答。值得一提的是,本例和下一例的解在域上的最大幅值达到 10 000 以上,采用 $T = 0.005$ 的绝对误差限是相当苛刻的,然而本文方法仍然能够给出全面严格满足误差限的有限元解。

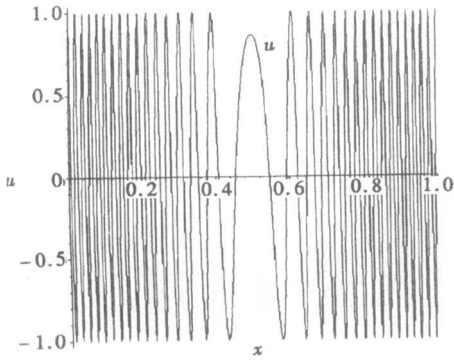


图 12 u 的分布示意图

例 5 控制方程及边界条件同式(3),令 $u = e^{10x} \sin 10\pi x$, f 由微分方程导出。 u 的分布如图 11 所示,是一个具有振荡性的问题,且函数变化剧烈。只采用消差法求解。表 8 是线性元至 4 次元求解的结果。可见,预设的误差限得到了全面严格的满足。网格结点分布大体上也是左疏右密。

例 6 控制方程同式(3),令 $u = \cos(100\sin(\pi x))$, f 和两端位移边界条件由 u 导出。 u 的分布如图 12 所示,是一个具有高度振荡性的问题。只采用消差法求解。表 9 给出了线性元至 4 次元求解的结果。可见,预设的误差限得到了全面严格的满足。网格结

点分布在中间相对较疏,两端相对较密。

5 结 语

理论分析和大量数值算例表明,本文提出的自适应方法,对于大多数问题,一步自适应求解即可得到基本满意的结果,对于高度振荡问题也只需增加一两次自适应计算,且生成的网格同对精确解做相应的自适应插值得到的网格几乎完全一致。此外,本法对高次元也同样适用,从算例来看,相比之下二次元表现比较占优:冗余精度较少,单元数适中。总之,本文提出的自适应求解方法,基本上实现了第 3 节中预定的目标,是一个简单、实用、高效、可靠的自适应求解方法。对于该法更加高效实用的数值实施和技巧,将在另文中介绍。

[参 考 文 献]

- [1] Babuska I, Rheinboldt W C. A posteriori error analysis of finite element method for one-dimensional problems[J]. SIAM Journal on Numerical Analysis, 1981, 18(3): 565—589.
- [2] Zienkiewicz O C, Zhu J Z. The superconvergence patch recovery (SPR) and a posteriori error estimates, part 1: the recovery technique[J]. Internat J Numer Methods Engrg, 1992, 33(7): 1331—1364.
- [3] Zienkiewicz O C, Zhu J Z. The superconvergence patch recovery (SPR) and a posteriori error estimates, part 2: error estimates and adaptivity[J]. Internat J Numer Methods Engrg, 1992, 33(7): 1365—1382.
- [4] 林群,朱起定.有限元的预处理和后处理理论[M].上海:上海科学技术出版社,1994.
- [5] 陈传森.有限元超收敛构造理论[M].长沙:湖南科学技术出版社,2002.
- [6] Ascher U, Christiansen J, Russell R D. Algorithm 569, COLSYS: Collocation software for boundary value ODEs[D2]. ACM Trans Math Software, 1981, 7(2): 223—229.
- [7] YUAN Si. The Finite Element Method of Lines[M]. Beijing, New York: Science Press, 1993.
- [8] 袁驷,王枚,林永静,等.有限元(线)法超收敛应力计算的新思路[J].工程力学(增刊).长沙:全国结构工程大会,2002, 112—118.
- [9] 袁驷,王枚.一维有限元后处理超收敛解答计算的 EEP 法[J].工程力学,2004, 21(2): 1—9.

- [10] Strang G, Fix G. An Analysis of the Finite Element Method [M]. London: Prentice_Hall, 1973.
- [11] Douglas J, Dupont T. Galerkin approximations for the two point boundary problems using continuous piecewise polynomial spaces[J]. Numer Math, 1974, 22(2): 99—109.

Self_Adaptive Strategy for One_Dimensional Finite Element Method Based on EEP Method

YUAN Si, HE Xue_feng

(Department of Civil Engineering, Tsinghua University, Beijing 100084, P. R. China)

Abstract: Based on the newly_developed element energy projection (EEP) method for computation of super_convergent results in one_dimensional finite element method (FEM), the task of self_adaptive FEM analysis was converted into the task of adaptive piecewise polynomial interpolation. As a result, a satisfactory FEM mesh can be obtained, and further FEM analysis on this mesh would immediately produce an FEM solution which usually satisfies the user specified error tolerance. Even though the error tolerance was not completely satisfied, one or two steps of further local refinements would be sufficient. This strategy has been found to be very simple, rapid, cheap and efficient. Taking the elliptical ordinary differential equation of the second order as the model problem, the fundamental idea, implementation strategy and detailed algorithm were described. Representative numerical examples are given to show the effectiveness and reliability of the proposed approach.

Key words: FEM; self_adaptive solution; super_convergence; element energy projection; ODE