Univ. u Novom Sadu
Zb. Rad. Prirod.–Mat. Fak.
Ser. Mat. 24, 1 (1994), 331-340

Review of Research
Faculty of Science
Mathematics Series

ON MESH GENERATION FOR SINGULAR PERTURBATION PROBLEMS

Relja Vulanović
Institute of Mathematics, University of Novi Sad
Trg Dositeja Obradovića 4, 21000 Novi Sad, Yugoslavia

Zoran Ovcin
Faculty of Engineering, University of Novi Sad
Trg Dositeja Obradovića 6, 21000 Novi Sad, Yugoslavia

Abstract

A well known mesh generation technique is tested on singularly perturbed boundary value problems. Some additional steps are proposed in order to improve the accuracy of the numerical solution.

AMS Mathematics Subject Classification (1991): 65L10
Key words and phrases: boundary value problem, singular perturbation, finite–difference scheme, mesh generation.

1. Introduction

We shall consider the following linear singularly perturbed boundary value problem:

$$Lu := -\epsilon u'' - b(x)u' + c(x)u = f(x)\,,\quad x \in I = [0,1]\,,$$
$$u(0) = U_0\,,\; u(1) = U_1\,, \tag{1}$$

with a small positive perturbation parameter ϵ, sufficiently smooth functions b , c and f, and given numbers U_- , U_+. We shall assume that at least one of the following conditions hold for $x \in I$:

$$c(x) \geq c_* > 0 \; ; \tag{2}$$

$$(\; b(x) \geq 0 \;\text{ or }\; b(x) \leq 0 \;) \;\text{ and }\; |b(x)| + c(x) \geq g_* > 0. \tag{3}$$

These conditions are the same as in [5] and they guarantee that there exists a unique solution u to the problem (1). We shall solve the problem numerically by a finite–difference method from [5]. The conditions (2) and (3) also guarantee the stablity uniform in ϵ and the unique solvability of the corresponding discrete problem.

It is well known that u may have several interior and boudary layers and because of that it is natural to discretize the problem on non–equidistant meshes which are dense in the layers. For problem (1), with condition (2) at least, such meshes can be given in advance since it is possible to locate the layers and to describe the behaviour of u by estimating its derivatives, see [1]. This is the simplest mesh construction method, but we shall be interested in mesh generation which does not require any a priori information on u, on the position of the layers etc. The reason for this is that we plan to apply the technique to quasilinear problems:

$$-\epsilon u'' - b(x,u)u' + c(x,u) = 0 \; ,$$

for which it is generally difficult to locate the layers in advance and to estimate the derivatives of u. Thus, linear problem (1) is chosen only for the reasons of simplicity. We want to test the method on the simplest singular perturbation problems before moving to the more complicated ones. The mesh generation which we shall apply is based on the procedure from [3]. However, we shall add some steps from the procedures from [2] and [4], and in this way we shall improve numerical results considerably.

It is quite easy to apply the method to semilinear problems:

$$-\epsilon u'' - b(x)u' + c(x,u) = 0.$$

For the quasilinear problems, however, a diffrent scheme should be used, and we expect to encounter some additional difficulties.

The discretization scheme will be given in Section 2. Then we shall describe the mesh generation procedure in Section 3, and end with numerical results in Section 4.

2. Discretization

One of the two schemes from [5] will be used to discretize the problem (1). For the sake of completeness, we shall repeat the scheme here.

Let I_h be an arbitrary non–equidistant mesh with the mesh points:

$$0 = x_0 < x_1 < \ldots < x_n = 1 \ , \ \ n \in \mathbf{N},$$

and let:

$$h_i = x_i - x_{i-1} \ , \ i = 1, 2, \ldots, n \ ,$$

$$x_{i+1/2} = \frac{x_i + x_{i+1}}{2} \ , \ i = 1, 2, \ldots, n \ .$$

By w_i , $i = 0, 1, \ldots, n$, we shall denote an arbitrary mesh function on I^h. Let us introduce the following finite–difference operators:

$$D''_C w_i = \frac{2\,(h_{i+1} w_{i-1} - (h_i + h_{i+1}) w_i + h_i w_{i+1})}{h_i h_{i+1} (h_i h_{i+1})} \ ,$$

$$D'_C w_i = \frac{w_{i+1} - w_{i-1}}{h_i + h_{i+1}} \ ,$$

$$D'_U w_i = \frac{(s_i + |s_i|)(w_{i+1} - w_i)}{h_{i+1}} + \frac{(s_i - |s_i|)(w_i - w_{i-1})}{h_i} \ , \ s_i = \frac{1}{2} sg(b(x_i)) \ ,$$

$$D'_M w_{i+1/2} = \frac{w_{i+1} - w_i}{h_{i+1}} \ ,$$

$$D^0_{M+} w_{i+1/2} = \frac{3 w_i - w_{i-1}}{2} \ ,$$

$$D^0_{M-} w_{i-1/2} = \frac{3 w_i - w_{i+1}}{2} \ ,$$

(the notation $w_{i\pm 1/2}$ should be understood formally). Using these operators we define the discretization schemes:

$$L^h_U w_i := -\epsilon\, D''_C w_i - b(x_i) D'_U w_i + c(x_i) w_i \ ,$$

$$L^h_C w_i := -\epsilon\, D''_C w_i - b(x_i) D'_C w_i + c(x_i) w_i \ ,$$

$$L^h_{M\pm} w_{i\pm 1/2} := -\epsilon\, D''_C w_i - b(x_{i\pm 1/2}) D'_M w_{i\pm 1/2} + c(x_{i\pm 1/2}) D^0_{M\pm} w_{i\pm 1/2} \ .$$

Finally, we combine the schemes and obtain the following discretization of (1):

$$L^h w_0 := w_0 = U_0 \ ,$$
$$L^h w_i := \begin{cases} L_C^h w_i = f(x_i) \ , & \text{if } -\frac{1}{h_{i+1}} < \rho_i < \frac{1}{h_i} \ , \\ L_{M+}^h w_{i+1/2} = f(x_{i+1/2}) \ , & \text{if } \rho_i \geq \frac{1}{h_i} \\ & \text{and } b(x_{i+1/2}) \geq 0 \ , \\ L_{M-}^h w_{i-1/2} = f(x_{i-1/2}) \ , & \text{if } \rho_i \leq -\frac{1}{h_{i+1}} \\ & \text{and } b(x_{i-1/2}) \leq 0 \ , \\ L_U^h w_i = f(x_i) \ , & \text{otherwise} \ , \end{cases}$$
$$i = 1, 2, \ldots, n-1 \ ,$$
$$L^h w_n := w_n = U_1 \ ,$$

where

$$\rho_i = b(x_i)/2\epsilon \ , \ \ i = 1, 2, \ldots, n \ .$$

It was shown in [5] that the discretization scheme L^h (which was denoted by L_2^h in [5]) is stable uniformly in ϵ (provided (2) or (3) holds), and that it is second order accurate on locally quasi–equidistant meshes. It shares these properties with the other scheme (L_1^h) from [5], but it is simpler and for that reason we have chosen it for our numerical experiments.

3. Mesh Generation

The mesh generation will be explained by the algorithm followed by the description of its steps.

3.1. Algorithm

```
/* INITIALIZATION */
mesh := equidistant;
solve_discrete_problem;
/* THE LOOP */
REPEAT
        equidistribution;
        solve_discrete_problem
UNTIL exit_criterion;
```

*/ FINALIZATION */
insert_mesh_points;
smoothing;
solve_discrete_problem;

3.2. Initialization Steps

These are the standard steps: the discrete problem from the previous section is solved on an equidistant mesh ($x_i^0 = i/n, \;\; i = 0, 1, \ldots, n$). Let us denote the obtained numerical solution by $w_i^0, \;\; i = 0, 1, \ldots, n$.

3.3. Loop

Let x_i^{k-1} and w_i^{k-1} $(i = 0, 1, \ldots, n)$ be the mesh and the numerical solution, respecively, obtained in the $(k-1)$-th step $(k = 1, 2, \ldots)$. Then in the k-th step, the polygon

$$\{(x_i^{k-1}, w_i^{k-1}) \, , \; i = 0, 1, \ldots, n\}$$

is formed and its lenght l is calculated:

$$l := \sum_{i=1}^{n} l_i \, , \; l_i := \sqrt{(x_i^{k-1} - x_{i-1}^{k-1})^2 + (w_i^{k-1} - w_{i-1}^{k-1})^2} \, .$$

Let

$$\delta := l/n \, .$$

This average lenght is used to devide the polygon into n successive parts (also polygonal lines), each of lenght δ. Let $\{(x_i^k, \overline{w}_i^k) \, , \; i = 0, 1, \ldots, n\}$ be the endpoints of the parts. Then $\{x_i^k \, , \; i = 0, 1, \ldots, n \}$ is the new mesh on which the discrete problem from Section 2 is to be solved. In this way we obtain the new numerical solution $w_i^k, \;\; i = 0, 1, \ldots, n$, and the new cycle can be started with the polygon $\{(x_i^k, w_i^k) \, , \; i = 0, 1, \ldots, n\}$. If the discrete problem is solved by iteration (which will always be the case for nonlinear problems), then $\overline{w}_i^k \, , \; i = 0, 1, \ldots, n$, may be used as the initial guess.

The cycle is repeated until the exit criterion is satisfied. First of all, we bound the number of cycles:

$$k_{min} \leq k \leq k_{max},$$

where k_{min} and k_{max} are given positive integers. Furthermore, let

$$\eta_{max} := \max_{1\le i\le n} \eta_i \ , \ \eta_i := |l_i - \delta| \ ,$$

and let γ be a given positive parameter. Then the exit criterion reads:

$$\eta_{max} \le \gamma\delta \ \text{ and } \ k_{min} \le k$$

or

$$k = k_{max} \ .$$

The steps described represent a discrete variant of the approach taken by White in [6]. Essentially the same procedure was used in [3] for one–dimensional parabolic problems. The finalization steps are not part of the mesh generation procedures from [6] and [3]. They are introduced here in order to improve the accuracy of the numerical solution.

3.4. Finalization Steps

These steps are used in the procedures from [2] and [4] as a part of the mesh generating cycle. Here, we use them only once to finish the procedure. First we insert some additional mesh points. We choose a positive integer n_* to monitor the total number of new mesh points inserted. Then p_i mesh points are inserted uniformly spaced into (x_{i-1}, x_i), $i = 1, 2, \dots, n$:

$$p_i := int(\frac{\eta_i}{l} n_*) \ ,$$

where $int(s)$ denotes the non–negative integer closest to s. Thus, the greater the difference between l_i and δ (and we can expect this in the layers), the greater the number of the mesh points inserted in (x_{i-1}, x_i).

This step may result in a mesh with abrubt changes of the local mesh step size, which may reduce the accuracy of the discretization scheme. Because of that, the smoothing procedure is introduced. It inserts new mesh points as well, but in a different manner. We consider the mesh smooth if the following holds for all indices i:

$$\frac{1}{Q} \le \frac{h_{i+1}}{h_i} \le Q \ ,$$

where $Q > 1$ is given. This is the exit criterion of the smoothing procedure. Until it is satisfied we repeat the following steps:

$$\text{if } h_{i+1}/h_i > Q\,, \quad \text{then add } x_{i+1/2} \text{ as a new mesh point;}$$

$$\text{if } h_i/h_{i+1} > Q\,, \quad \text{then add } x_{i-1/2} \text{ as a new mesh point.}$$

4. Numerical Results

Let us make a comparison between the mesh generating procedures from the previous section with and without the finalization step. Let n_f denote the final number of mesh steps, let x_i , $i = 0, 1, \ldots, n_f$, be the final mesh points, and let w_i , $i = 0, 1, \ldots, n_f$, be the numerical solution of the final mesh. In case of the procedure without finalization, we have $n = n_f$. If finalization is applied, then it holds that $n_f \geq n$.

Three test problems will be considered. For all of them the exact solution u is known, and in all the three cases there is one boundary layer at $x = 0$. The layers are of different types, though, and we find the problems quite suitable to illustrate the possibilities of the mesh generation procedures. The errors

$$\text{max\_err} = \max_{0 \leq i \leq n_f} |u(x_i) - w_i| \,.$$

will be presented in tables. In case of the mesh generation with finalization, they will be followed by n_f in parentheses. In what follows, the problem will be given with u which determines $f(x)$ and U_- , U_+. Then we shall present the mesh generating parameters and finally the tables. The entries marked by * denote the mesh generating procedure which has been stoped because of $k = k_{max}$.

We can conclude from the results that the finialization steps pay off since the extra effort is not so big, and the errors are considerably better, particularly for smaller values of ϵ. Furthermore, the results are better than those from [4], where different mesh adaption cycles, similar to the ones from [2], were used.

4.1. First Example

$$-\epsilon u'' - x u' + 2u = f(x)\,, \quad u = x + e^{-\frac{x^2}{\epsilon}}$$

$$k_{min} = 2\ ,\quad k_{max} = 12\ ,\quad \gamma = 1.0$$

finalization with $n_* = 10$, $Q = 3.0$

	$n = 15$	$n = 40$	$n = 90$	$n = 190$
ϵ	max_err			
1.0E-02	0.00271 (25)	0.00094 (50)	0.00042 (96)	0.00012 (195)
1.0E-04	0.01836 (29)	0.00497 (50)	0.00203 (97)	0.00052 (193)
1.0E-06	0.02318 (29)	0.00519 (51)	0.00354 (99)	0.00124 (196)
1.0E-08	0.01262 (34)	0.00579 (58)	0.00221 (100)	0.00132 (205)
1.0E-12	0.02247 (42)	0.00630 (65)	0.00216 (112)	0.00119 (206)

no finalization

	$n = 25$	$n = 50$	$n = 100$	$n = 200$
ϵ	max_err			
1.0E-02	0.00629	0.00182	0.00045	0.00011
1.0E-04	0.04668	0.01244	0.00794	0.00178
1.0E-06	0.08635	0.05544	0.03110	0.01189
1.0E-08	0.07338	0.07316	0.03516	0.00983
1.0E-12	0.08033	0.08246	0.03783	0.03701

4.2. Second Example

$$-\epsilon u'' - u = 0\ ,\quad u = 1 - e^{-\frac{x}{\epsilon}}$$

$$k_{min} = 2\ ,\quad k_{max} = 12\ ,\quad \gamma = 1.0$$

finalization with $n_* = 10$, $Q = 3.0$

	$n = 15$	$n = 40$	$n = 90$	$n = 190$
ϵ	max_err			
1.0E-02	0.00148 (29)	0.00323 (46)	0.00032 (98)	0.00027 (195)
1.0E-04	0.00232 (35)	0.00185 (56)	0.00141 (99)	0.00093 (196)
1.0E-06	0.00405 (41)	0.00442 (60)	0.00064 (106)	0.00126 (202)
1.0E-08	0.00618 (47)	0.00145 (72)	0.00073 (112)	0.00068 (209)
1.0E-12	0.00588*(61)	0.00126 (80)	0.00053 (130)	0.00037 (223)

no finalization

	$n = 25$	$n = 50$	$n = 100$	$n = 200$
ϵ	max_err			
1.0E-02	0.03817	0.01298	0.00111	0.00032
1.0E-04	0.02205	0.03498	0.01768	0.01007
1.0E-06	0.04758	0.04410	0.01750	0.00997
1.0E-08	0.03470	0.03448	0.01783	0.00997
1.0E-12	0.02842	0.03402	0.01972	0.00997

4.3. Third Example

$$-\epsilon u'' - u' = -(\epsilon + 1)e^x \ , \ \ u = -e^{-\frac{x}{\epsilon}} + e^x$$

$$k_{min} = 2 \ , \ \ k_{max} = 12 \ , \ \ \gamma = 0.75$$

finalization with $Q = 3.0$

	$n = 15$ $n_* = 10$	$n = 30$ $n_* = 20$	$n = 60$ $n_* = 40$	$n = 120$ $n_* = 80$
ϵ	max_err			
1.0E-02	0.00407 (30)	0.00108 (53)	0.00045 (106)	0.00011 (209)
1.0E-04	0.00454 (34)	0.00229 (58)	0.00139 (109)	0.00029 (207)
1.0E-06	0.00339 (43)	0.00262 (65)	0.00110 (117)	0.00022 (216)
1.0E-08	0.00538 (50)	0.00757*(71)	0.00244 (123)	0.00122 (222)
1.0E-12	0.02282*(53)	0.00415*(86)	0.00131 (132)	0.00135 (227)

no finalization

	$n = 25$	$n = 50$	$n = 100$	$n = 200$
ϵ	max_err			
1.0E-02	0.03272	0.01007	0.00096	0.00032
1.0E-04	0.02164	0.03626	0.00512	0.00950
1.0E-06	0.02535	0.03033	0.00475	0.00660
1.0E-08	0.03306	0.00555*	0.00644	0.00904
1.0E-12	0.00426*	0.02766	0.00471	0.00726

References

[1] Liseikin, V. D., Petrenko, V. E., On numerical solution of nonlinear singularly perturbed problems. Siberian Branch of the USSR Acad. of Sci., Computing Center, Priprint No. 687, Novosibirsk, 1987.

[2] Kreiss, B., Kreiss, H.-O., Numerical methods for singular perturbation problems. SIAM J. Numer. Anal. 18 (1981), 262-275.

[3] Sanz–Serna, J. M., Christie, I., A simple adaptive technique for nonlinear wave problems. J. Comput. Physics 67 (1986), 348-360.

[4] Vulanović, R., Mesh Construction for Discretization of Singularly Perturbed Boundary Value Problems. Ph. D. Thesis, University of Novi Sad, 1986.

[5] Vulanović, R., Non–equidistant generalization of the Guschin–Schennikov scheme. Z. angew. Math. Mech. 67 (1987), 625-632.

[6] White, A. B., Jr., On selection of equidistributing meshes for two-point boundary-value problems. SIAM J. Numer. Anal. 16 (1979), 472-501.

REZIME

O GENERISANJU MREŽA ZA SINGULARNO PERTURBOVANE PROBLEME

Jedna poznata tehnika za generisanje mreža diskretizacije je testirana na singularno perturbovanim konturnim problemima. Predloženi su neki dodatni koraci da bi se poboljšala tačnost numeričkog rešenja.

Received by the editors September 15, 1991