CHAPTER 19

A Domain Decomposition Method for Boundary Layer Problems*

Garry Rodrigue†
Edna Reiter††

Abstract: In this paper we analyze the behavior of a specific domain decomposition technique for solving a boundary value problem of the type

$$L_\varepsilon[u] = \frac{\partial}{\partial x}u + \varepsilon\Delta u = 0, \quad (x, y) \in \Omega$$

We are concerned primarily in problems where ε is sufficiently small and where the boundary conditions yield ordinary and parabolic boundary layers in the solution. The global domain is decomposed into subdomains according to the particular layers and the global solution is obtained by piecing together the different subdomain solutions. An algorithm for locating the layers and consequently the subdomains will be constructed. Numerical results will be presented.

1 Introduction

In this paper we study the application of a domain decomposition technique to the convection-diffusion equation

$$\frac{\partial u}{\partial x} - \varepsilon\left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right] = 0 \tag{1.1}$$

where $0 < \varepsilon << 1$ and $(x, y) \in \Omega = [0, 1] \times [0, 1]$. Given certain boundary conditions on (1.1), the solution is known to possess boundary layers where, for example, there can be regions of Ω such that

* This work was performed under the auspices of the U.S. Department of Energy at the Lawrence Livermore Laboratory under Contract W-7405-Eng-48.

† University of California, Davis, and Lawrence Livermore National Laboratory, Livermore, California 94550

‡ Department of Mathematics, California State University, Hayward, California 94542

$$\begin{aligned} u &\simeq \textstyle\sum_{n=0}^{m} \varepsilon^n \phi_n \\ \frac{\partial \phi_0}{\partial x} &= 0 \\ \frac{\partial \phi^n}{\partial x} &= \varepsilon \left[\frac{\partial^2 \phi_{n-1}}{\partial x^2} + \frac{\partial^2 \phi_{n-1}}{\partial y^2} \right], n > 0 \end{aligned}$$

(called ordinary boundary layers) or there can be regions such that

$$\begin{aligned} u &\simeq \textstyle\sum_{n=0}^{\ell} \varepsilon^n \Psi_n \\ \frac{\partial \Psi_0}{\partial x} - \frac{\partial^2 \Psi_0}{\partial \xi^2} &= 0 \\ \frac{\partial \Psi_n}{\partial x} - \frac{\partial^2 \Psi_n}{\partial \xi^2} &= \varepsilon \frac{\partial^2 \Psi_{n-1}}{\partial x^2}, \ n > 0 \end{aligned}$$

(called parabolic boundary layers) and ξ is a local "stretched" coordinate, [1] .

In the following sections of the paper, we will define a domain decomposition on Ω based on the boundary layer behavior of the solution and then apply a variant of the Schwarz Alternating Procedure to obtain a numerical approximation of the solution of (1.1)

2 Parabolic Layers

In this paper, the boundary conditions we impose on (1.1) are

$$\begin{aligned} &(a) \quad u(x,1) = 1, \qquad 0 \le x \le 1, \\ &(b) \quad u(x,0) = 0, \qquad 0 \le x \le 1, \\ &(c) \quad u(0,y) = 1, \qquad 0 < y < 1, \\ &(d) \quad \frac{\partial u}{\partial x}(1,y) = 0, \quad 0 < y < 1. \end{aligned} \tag{2.1}$$

In [2], it was established that the sequence of functions $\{u^{(n)}\}$ defined by the iteration

$$\begin{cases} \dfrac{\partial u^{(0)}}{\partial x} - \varepsilon \dfrac{\partial^2 u^{(0)}}{\partial^2 y} = 0 \\ \dfrac{\partial u^{(n)}}{\partial x} - \varepsilon \dfrac{\partial^2 u^{(n)}}{\partial y^2} = \varepsilon \dfrac{\partial^2 u^{(n-1)}}{\partial x^2}, \ n > 0 \end{cases} \tag{2.2}$$

$$(2.2a)\quad u^{(n)}(x,1) = 1, \quad 0 \le x \le 1$$

$$(2.2b)\quad u^{(n)}(x,0) = 0, \quad 0 \le x \le 1$$

$$(2.2c)\quad u^{(n)}(\alpha,y) = g(y), \quad 0 < y < 1, \ \alpha = \text{fixed}$$

satisfies

$$\|u - u^{(n)}\|_\infty = O(\varepsilon^{n+1})$$

when

$$(2.3)\qquad \frac{d^k g}{\partial y^k}(0) = \frac{d^k g}{\partial y^k}(1) = 0, \quad k = 0,1,\ldots,n.$$

Since the boundary conditions (2.1a-c) do not satisfy (2.3), a domain decomposition strategy would be to split Ω into the subregions $\Omega = \Omega_1 \bigcup \Omega_2$ where

$$\begin{aligned} \Omega_1 = &\ [0,\ell_1]\times[0,1], \quad \ell_1 > 0, \\ \Omega_2 = &\ [\ell_2,1]\times[0,1], \\ \ell_2 < &\ \ell_1, \end{aligned}$$

(see Figure 1) and numerically solve (1.1)—(2.1a–d) on Ω_1. This is followed by numerically carrying out the iteration (2.2) with conditions (2.2a-c) and using

$$(2.4)\qquad g(y) = u(\ell_2, y)$$

where u is the computed solution in Ω_1.

To test the feasibility of this approach, we take several values of $\ell_1 > 0$ and for each of these values we solve (1.1) with $\varepsilon = .0005$, the boundary conditions (2.1a-c) and

$$(2.5)\qquad \frac{\partial u}{\partial x}(\ell_1, y) = 0, \quad 0 < y < 1.$$

We then carry out the iteration (2.2) on $\Omega_2 = [\ell_2, 1]\times[0,1]$, $\ell_2 = \ell_1 - \Delta x$, and

$$\begin{aligned} u^{(n)}(x,1) &= 1, & \ell_2 \le x \le 1 \\ u^{(n)}(x,0) &= 0, & \ell_2 \le x \le 1 \\ u^{(n)}(\ell_2,y) &= u(\ell_2,y), & 0 \le y \le 1. \end{aligned}$$

On Ω_1, the equation (1.1) is approximated by 2nd-order finite differences on a grid with $\Delta x = .0015$ and $\Delta y = .001$ and the resultant linear system is solved with a direct matrix solver. (2.2), on the other hand, requires the solution of inhomogeneous heat equations with the x-variable interpreted as the time variable. In this situation, the Crank-Nicholson method is used on a grid with $\Delta x = .01, \Delta y = .001$. Table 1 lists the results. Each entry of the table is $\|U^{(n)} - U^{(n-1)}\|_\infty$ where

$U^{(n)}$ is the computed approximation to $u^{(n)}$. Note that convergence in all cases is quite rapid.

In [2], it is established that for fixed Δx and Δy, there exists $\varepsilon > 0$ so that divergence occurs. In order to determine such values,we use the same grid structure as before, take $\ell_2 = 0.041$, vary the ε and list the errors in Table 2. As can be seen, larger values of ε will result in divergence.

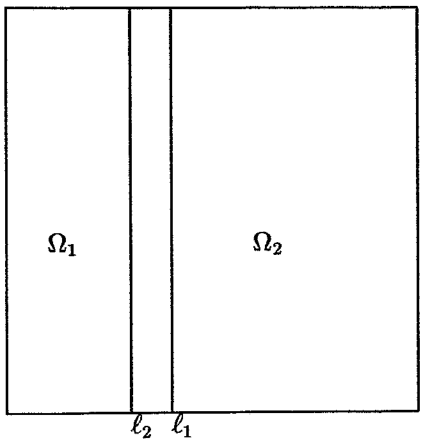

Figure 1

Table 1:

$n\backslash\ell_1$	.003	.018	.033	.048	.063
1	.90401	.77714	.71635	.67007	.63575
2	.01379	.00287	.00143	.00100	7.8×10^{-4}
3	.00601	1.15×10^{-4}	4.3×10^{-5}	2×10^{-5}	1.3×10^{-5}
4	3.9×10^{-5}	6.1×10^{-6}	2.08×10^{-6}	8.9×10^{-7}	5×10^{-7}
5	3.1×10^{-6}	3.8×10^{-7}	1.2×10^{-7}	5×10^{-8}	2×10^{-8}
6	2.8×10^{-7}	3×10^{-8}	1×10^{-8}		
7	2×10^{-8}				

Table 2:

$n\backslash\varepsilon$	5×10^{-5}	5×10^{-4}	5×10^{-3}	5×10^{-2}
1	.73911	.63575	.277827	.037807
2	.000178	.00078	.010518	0.8499
3	4.1×10^{-7}	1.3×10^{-5}	.00195	2.6304
4	3×10^{-9}	5×10^{-7}	.00085	21.944
5		2×10^{-8}	4.6×10^{-4}	184.6
6			2.9×10^{-4}	∞

3 Ordinary Layers

We consider the problem (1.1) with boundary conditions (2.1a-d). In [3] it was established that the sequence of functions $u^{(n)}$ defined by the iteration

$$\begin{aligned} &(a) \quad u^{(0)}(x,y) &&= 1 \\ (3.1) \qquad &(b) \quad \frac{\partial u^{(n)}}{\partial x} &&= \varepsilon\left[\frac{\partial^2 u^{(n-1)}}{\partial x^2} + \frac{\partial^2 u^{(n-1)}}{\partial y^2}\right], \; n > 0 \\ &(c) \quad u^{(n)}(x,0) &&= f(x), 0 \leq x \leq 1 \end{aligned}$$

$$(3.2) \qquad \frac{d^k f}{dx^k}(0) = 1, \; k = 0, 1, \ldots, n,$$

satisfies

$$\|u - u^{(n)}\|_\infty = O(\varepsilon^{n-1})$$

(u is the solution to (1.1)).

As before, since the boundary conditions (2.1c) does not satisfy (3.2), a domain decomposition strategy would be to split Ω into the subregions $\Omega = \Omega_1 \bigcup \Omega_2$ where

$$(3.3) \qquad \begin{aligned} \Omega_1 &= [0,1] \times [0, \ell_1], \; \ell_1 > 0 \\ \Omega_2 &= [0,1] \times [\ell_2, 1], \; \ell_2 < \ell_1, \end{aligned}$$

(see Figure 2) and numerically solve (1.1) on Ω_1, followed by numerically carrying out the iteration (3.1a-c) with

$$(3.4) \qquad f(x) = u(x, \ell_2)$$

where u is computed solution in Ω_1

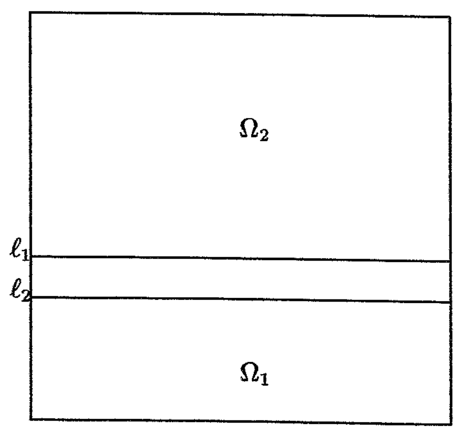

Figure 2

To test the feasibility of this approach, we take several values of $\ell_2 > 0$ and solve (3.1a-c)—(3.4). As a test case, we take $\varepsilon = 2\times10^{-4}$, use the exact solution of (1.1) on Ω_1 (cf. [2]) and then carry out (3.1a–c) -(3.4) on Ω_2 using a Backward Euler method on a grid with $\Delta x = \Delta y = 10^{-2}$. Convergence to an error of $\|U^{(n)}-U^{(n-1)}\|_\infty < 10^{-5}$ occurred at 30 iterations for each value of $\ell_2 = 10^{-2}, 10^{-1}, 4\times10^{-1}$. Also, for $\ell_2 = 10^{-2}$, convergence occurred for $\varepsilon = 10^{-4}$ whereas divergence occurred for $\varepsilon = 5\times10^{-4}$.

4 Schwarz Method

In this section we develop a Schwarz Alternating Procedure for solving (1.1) based on the results of the previous two sections. That is, we split $\Omega = \Omega_1 \bigcup \Omega_2 \bigcup \Omega_3$ where

$$\begin{aligned}\Omega_1 &= [0,1]\times[b,1]\\ \Omega_2 &= [0,r]\times[0,t] \quad ,b<t,\\ \Omega_3 &= [\ell,1]\times[0,t] \quad ,\ell<r,\end{aligned}$$

(see Figure 3). Let $u_1^{(0)} = u_2^{(0)} = u_3^{(0)} = 1$, on $\Omega_1, \Omega_2, \Omega_3$ respectively. We then define the sequences $\{u_1^{(i)}\}, \{u_2^{(i)}\}, \{u_3^{(i)}\}$ as follows: for $i = 1, 2, \ldots,$

1) $u_2^{(i)}$ solves (1.1) on Ω_2 with

$$\begin{aligned}u_2^{(i)} &= u_1^{(i)} \quad \text{on} \quad [0,r]\times\{t\}\\ u_2^{(i)} &= 1 \quad \text{on} \quad \{0\}\times[0,t]\\ u_2^{(i)} &= 0 \quad \text{on} \quad [0,r]\times\{0\}\\ \frac{\partial}{\partial x}u_2^{(i)} &= 0 \quad \text{on} \quad \{r\}\times[0,t]\end{aligned}$$

2) $u_3^{(i)}$ solves (2.2) on Ω_3 with

$$\begin{aligned}u_3^{(i)} &= u_2^{(i)} \quad \text{on} \quad \{\ell\}\times[0,t]\\ u_3^{(i)} &= u_1^{(i-1)} \quad \text{on} \quad [\ell,1]\times\{t\}\\ u_3^{(i)} &= 0 \quad \text{on} \quad [\ell,1]\times\{0\}\end{aligned}$$

3) $u_1^{(i)}$ solves (3.1b) on Ω_1 with

$$\begin{aligned}u_1^{(i)} &= u_2^{(i)} \text{ on } \quad [0,r]\times\{b\}\\ u_1^{(i)} &= u_3^{(i)} \text{ on } \quad [r,1]\times\{b\}\end{aligned}$$

We carried out the above iterations under different scenarios to examine its convergence behavior to the solution of (1.1). In all cases, the following mesh sizes were used:

$$\Omega_1 \ : \ \Delta x = \Delta y = 10^{-2}$$

$$\Omega_2 \ : \ \Delta x = \Delta y = 10^{-3}$$

$$\Omega_3 \ : \ \Delta x = 10^{-2}, \ \Delta y = 10^{-3}$$

The numerical method of solution was the same as that in sections 2 and 3.

In the first experiment, the lower boundary of Ω_1 is fixed and the boundary between Ω_2 and Ω_3 is varied. In this case

$$\varepsilon = 10^{-4}$$

$$b = .03$$

$$t = .031$$

$$r = 0.41$$

$$\ell = r - k \times 10^{-3}$$

Table 3 records the results.

In the second experiment, the boundary between Ω_2 and Ω_3 is fixed and the lower boundary of Ω_1 is varied. In this case,

$$\varepsilon = 2 \times 10^{-4}$$

$$t = 0.31$$

$$b = t - k \times 10^{-2}$$

$$r = 0.41$$

$$\ell = .04$$

Table 4 records the results.

In the final experiment, the boundaries of $\Omega_1, \Omega_2, \Omega_3$ are held fixed and the value of ε is varied. In this case,

$$b = 0.3$$

$$t = 0.31$$

$$r = 0.41$$

$$\ell = 0.30$$

Table 5 records the results.

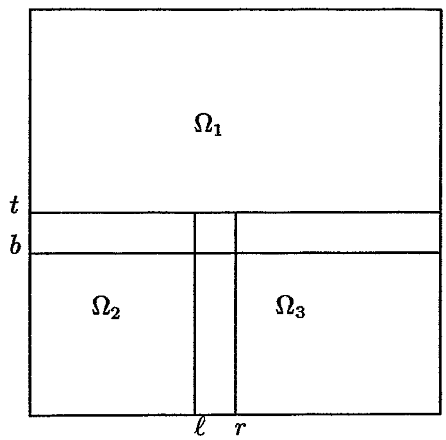

Figure 3

Table 3:

$n\backslash k$	1	.20	.40
1	.64325	.72092	.891205
2	.0022	.00213	.002195
3	5.26×10^{-4}	5.05×10^{-4}	4.53×10^{-4}
4	1.68×10^{-4}	1.59×10^{-4}	1.39×10^{-4}
5	5.5×10^{-5}	5.2×10^{-5}	4.3×10^{-5}
6	1.8×10^{-5}	1.6×10^{-5}	1.3×10^{-5}
7	5×10^{-6}	5×10^{-6}	4×10^{-6}
8	2×10^{-6}	1×10^{-6}	1×10^{-6}

Table 4:

$n\backslash k$	1	10	20
1	.2580	.3363	.5799
3	.6844	.2859	.0682
5	.9212	.4923	.0846
7	.7625	.4753	.0623
9	.4705	.3134	.0325
11	.2133	.1572	.0132
13	.0734		
15	.02		

Table 5:

$n\backslash \varepsilon$	5×10^{-5}	2×10^{-4}	2.5×10^{-4}	$3. \times 10^{-4}$	3.5×10^{-4}
1	.6684	.6718	.6659	.6561	.6432
2	10^{-4}	.0184	.0304	.0502	.0802
3	7×10^{-5}	.0146	.0428	.0978	.1907
4	8×10^{-7}	.0148	.0571	.1634	.3838
5	10^{-7}	.0129	.0658	.2344	.6615
6	10^{-8}	.0101	.0676	.2986	1.0091
7		.0072	.0629	.3437	∞
8		.0047	.0536	.3621	
9		.003	.0438	.3647	
10		.0018	.0337	.3459	
11		.001	.0241	.3044	

References

[1] Eckhaus, W., *Asymptotic Analysis of Singular Perturbations*, North-Holland, 1979

[2] Rodrigue, G., Hedstrom, G., Chin, C., and Reiter,E., *Domain Decomposition for Parabolic Boundary Layer Problems*; to appear

[3] Eckhaus, W. and de Jaeger, E.M., *Asymptotic Solutions of Singular Perturbation Problems for Linear Differential Equations of Elliptic type*, Arch. National Mech. Anal., Vol.23, p.p. 26-86