Theory & Computation of Singularly Perturbed Differential Equations

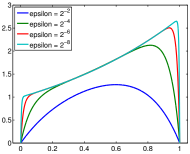
IIT (BHU) Varanasi, Dec 2017

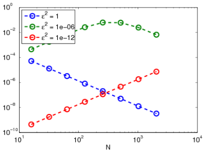
https://skumarmath.wordpress.com/gian-17/singular-perturbation-problems/

Niall Madden, NUI Galway

§2 Numerical methods and Uniform convergence

Version 03.12.17





Presentation version: not for printing

Outline

Monday, 4 December		
09:30 – 10.30	Registration and Inauguration	
10:45 – 11.45	1. Introduction to singularly perturbed problems	NM
12:00 – 13:00	2. Numerical methods and uniform convergence	NM
14:30 – 15:30	Tutorial (Convection diffusion problems)	NM
15:30 – 16:30	Lab 1 (Simple FEMs in MATLAB)	NM
Tuesday, 5 December		
09:30 – 10:30	3. Finite difference methods and their analyses	NM
10:45 – 11:45	4. Coupled systems of SPPDEs	NM
14:00 – 16:00	Lab 2 (Fitted mesh methods for ODEs)	NM
Thursday, 7 December		
09:00 – 10:00	8. Singularly perturbed elliptic PDEs	NM
10:15 – 11:15	9. Finite Elements in two and three dimensions	NM
01:15 – 15:15	Lab 4 (Singularly perturbed PDEs)	NM
Friday, 8 December		
09:00 – 10:00	10. Preconditioning for SPPs	NM

§2 Numerical methods and Uniform convergence

(60 minutes)

1. A reaction-diffusion problem
   - Uniform Convergence (heuristic)
   - A simple FDM
   - What is going wrong here?
2. A convection-diffusion problem
3. Uniform convergence
   - Layer resolving
4. References

Primary references

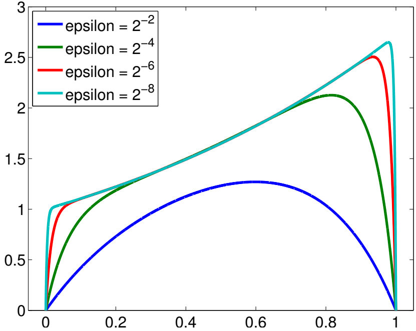
The definition of *parameter uniform*ity (Slides 16–18) is from [Farrell et al., 2000]. See also [Linß, 2010].

A reaction-diffusion problem

Let's recall our first example of a *singularly perturbed reaction-diffusion* equation.

$$-\varepsilon^2 u''(x) + b(x)u(x) = f(x), \qquad \text{on } \Omega = (0,1),$$

- ε is (still) a small parameter; it may take any value in $(0,1]$.
- There is $\beta > 0$ such that $b(x) \geqslant \beta > 0$.
- Boundary conditions: $u(0) = u(1) = 0$



In general, one must approximate the solutions to such problems by some numerical scheme.

A "*Parameter Robust*" or "*Uniformly Convergent*" method is one that yields an approximation U of u, such that one can prove an error estimate of the form

$$\|u - U\| \leqslant CN^{-p}$$

where C, p ("rate of convergence") are independent of the perturbation parameter ε, and discretization parameter N. This should be valid for all $\varepsilon \in (0, 1]$ and all N.

In particular, one should *not* have to assume that, for example, $N = \mathcal{O}(1/\varepsilon)$.

It is also desirable that any layer present should be resolved.

This explanation of "uniform convergence" is heuristic, (and we have not even specified $\|\cdot\|$). The concept will be will be made formal later.

The simplest numerical scheme one could apply to this problem is a second-order finite difference scheme on a uniform mesh.

- On the interval $\overline{\Omega} = [0, 1]$, form a uniform mesh with N intervals:

$$\Omega^N := \{x_i\}_{i=0}^N, \qquad \text{where } x_i = i/N = ih;$$

- Approximate $u''(x_i)$ as

$$u''(x_i) = \underbrace{\frac{1}{h^2}\big(u(x_{i-1}) - 2u(x_i) + u(x_{i+1})\big)}_{\delta^2 u(x_i)} + C\underbrace{\|u^{(4)}\|}_{\mathcal{O}(\varepsilon^{-4})} N^{-2}.$$

- Construct and solve the linear system

$$\mathbf{U}_0 = 0,$$

$$-\varepsilon^2\delta^2 U_i + b(x_i)U_i = f(x_i), \qquad i = 1, \dots, N-1$$

$$\mathbf{U}_N = 0.$$

If we implement the above finite difference method, and then calculate the maximum point-wise error, we get the following results.

$$\max_i |\mathbf{u}(x_i) - \mathbf{U}_i| \quad \text{where } u \text{ solves } -\varepsilon^2 u'' + u = e^x.$$

ε^2	$N = 64$	$N = 128$	$N = 256$	$N = 512$
1	7.447e-06	1.861e-06	4.654e-07	1.163e-07
10^{-2}	1.023e-03	2.568e-04	6.424e-05	1.607e-05
10^{-4}	7.689e-02	2.338e-02	6.192e-03	1.583e-03
10^{-6}	1.104e-02	4.203e-02	1.033e-01	9.666e-02
10^{-8}	1.113e-04	4.452e-04	1.779e-03	7.088e-03
10^{-10}	1.113e-06	4.453e-06	1.781e-05	7.125e-05
10^{-12}	1.113e-08	4.453e-08	1.781e-07	7.125e-07

We observe that,

- for small fixed N the error decreases as ε *decreases* (counter-intuitive)
- for small fixed ε, the error *increases* as N increases (i.e., not converging)

If we implement the above finite difference method, and then calculate the maximum point-wise error, we get the following results.

$$\max_i |\mathbf{u}(x_i) - \mathbf{U}_i| \quad \text{where } u \text{ solves } -\varepsilon^2 u'' + u = e^x.$$

ε^2	$N = 64$	$N = 128$	$N = 256$	$N = 512$
1	7.447e-06	1.861e-06	4.654e-07	1.163e-07
10^{-2}	1.023e-03	2.568e-04	6.424e-05	1.607e-05
10^{-4}	7.689e-02	2.338e-02	6.192e-03	1.583e-03
10^{-6}	1.104e-02	4.203e-02	1.033e-01	9.666e-02
10^{-8}	1.113e-04	4.452e-04	1.779e-03	7.088e-03
10^{-10}	1.113e-06	4.453e-06	1.781e-05	7.125e-05
10^{-12}	1.113e-08	4.453e-08	1.781e-07	7.125e-07

We observe that,

- for small fixed N the error decreases as ε *decreases* (counter-intuitive)
- for small fixed ε, the error *increases* as N increases (i.e., not converging)

If we implement the above finite difference method, and then calculate the maximum point-wise error, we get the following results.

$$\max_i |\mathbf{u}(x_i) - \mathbf{U}_i| \quad \text{where } u \text{ solves } -\varepsilon^2 u'' + u = e^x.$$

ε^2	$N = 64$	$N = 128$	$N = 256$	$N = 512$
1	7.447e-06	1.861e-06	4.654e-07	1.163e-07
10^{-2}	1.023e-03	2.568e-04	6.424e-05	1.607e-05
10^{-4}	7.689e-02	2.338e-02	6.192e-03	1.583e-03
10^{-6}	1.104e-02	4.203e-02	1.033e-01	9.666e-02
10^{-8}	1.113e-04	4.452e-04	1.779e-03	7.088e-03
10^{-10}	1.113e-06	4.453e-06	1.781e-05	7.125e-05
10^{-12}	1.113e-08	4.453e-08	1.781e-07	7.125e-07

We observe that,

- for small fixed N the error decreases as ε *decreases* (counter-intuitive)
- for small fixed ε, the error *increases* as N increases (i.e., not converging)

If we implement the above finite difference method, and then calculate the maximum point-wise error, we get the following results.

$$\max_i |\mathbf{u}(x_i) - \mathbf{U}_i| \quad \text{where } u \text{ solves } -\varepsilon^2 u'' + u = e^x.$$

ε^2	$N = 64$	$N = 128$	$N = 256$	$N = 512$
1	7.447e-06	1.861e-06	4.654e-07	1.163e-07
10^{-2}	1.023e-03	2.568e-04	6.424e-05	1.607e-05
10^{-4}	7.689e-02	2.338e-02	6.192e-03	1.583e-03
10^{-6}	1.104e-02	4.203e-02	1.033e-01	9.666e-02
10^{-8}	1.113e-04	4.452e-04	1.779e-03	7.088e-03
10^{-10}	1.113e-06	4.453e-06	1.781e-05	7.125e-05
10^{-12}	1.113e-08	4.453e-08	1.781e-07	7.125e-07

We observe that,

- for small fixed N the error decreases as ε *decreases* (counter-intuitive)
- for small fixed ε, the error *increases* as N increases (i.e., not converging)

If we implement the above finite difference method, and then calculate the maximum point-wise error, we get the following results.

$$\max_i |\mathbf{u}(x_i) - \mathbf{U}_i| \quad \text{where } u \text{ solves } -\varepsilon^2 u'' + u = e^x.$$

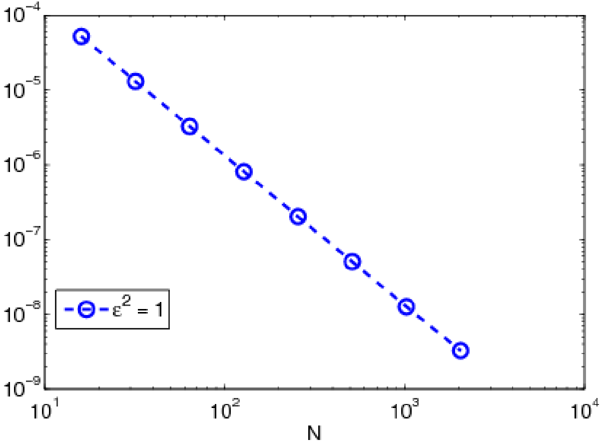
ε^2	$N = 64$	$N = 128$	$N = 256$	$N = 512$
1	7.447e-06	1.861e-06	4.654e-07	1.163e-07
10^{-2}	1.023e-03	2.568e-04	6.424e-05	1.607e-05
10^{-4}	7.689e-02	2.338e-02	6.192e-03	1.583e-03
10^{-6}	1.104e-02	4.203e-02	1.033e-01	9.666e-02
10^{-8}	1.113e-04	4.452e-04	1.779e-03	7.088e-03
10^{-10}	1.113e-06	4.453e-06	1.781e-05	7.125e-05
10^{-12}	1.113e-08	4.453e-08	1.781e-07	7.125e-07

We observe that,

- for small fixed N the error decreases as ε *decreases* (counter-intuitive)
- for small fixed ε, the error *increases* as N increases (i.e., not converging)

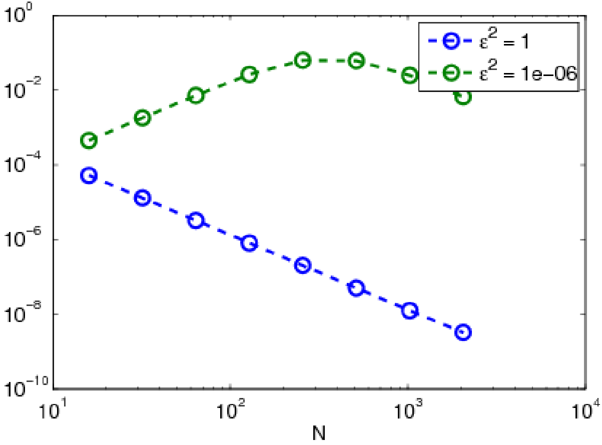
Comparing "convergence" for different values of ε.

$$\max_i |\mathbf{u}(x_i) - \mathbf{U}_i|$$



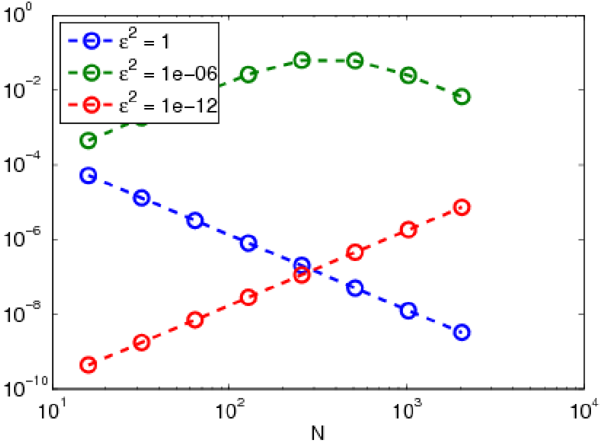
Comparing "convergence" for different values of ε.

$$\max_i |\mathbf{u}(x_i) - \mathbf{U}_i|$$



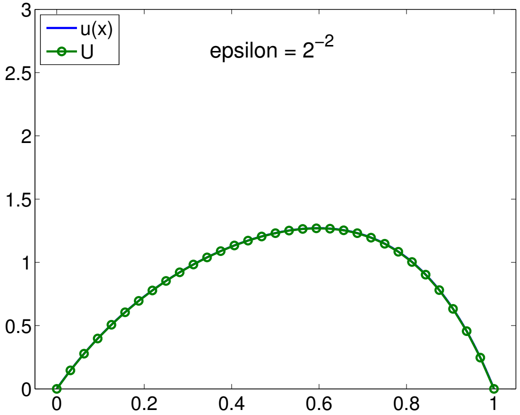
Comparing “convergence” for different values of ε.

$$\max_i |\mathbf{u}(x_i) - \mathbf{U}_i|$$



Why, for small fixed N, does the error appear to decrease as ε is reduced?

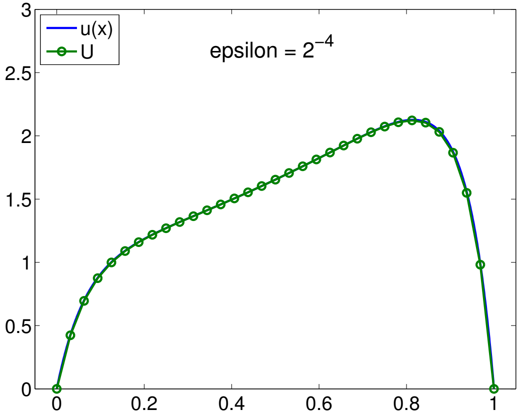
We fix $N = 32$ and take $\varepsilon = 10^{-2}, 10^{-4}, \dots, 10^{-10}$.



The pointwise errors are small because the layer is not resolved.

Why, for small fixed N, does the error appear to decrease as ε is reduced?

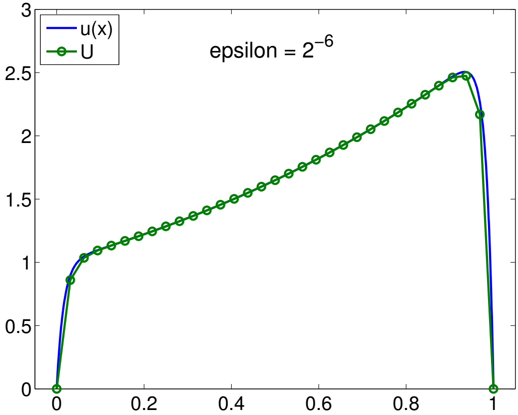
We fix $N = 32$ and take $\varepsilon = 10^{-2}, 10^{-4}, \ldots, 10^{-10}$.



The pointwise errors are small because the layer is not resolved.

Why, for small fixed N, does the error appear to decrease as ε is reduced?

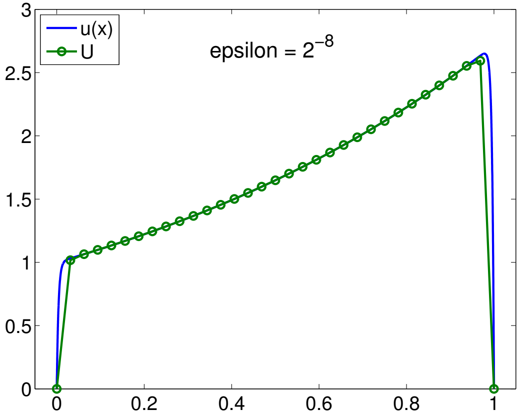
We fix $N = 32$ and take $\varepsilon = 10^{-2}, 10^{-4}, \ldots, 10^{-10}$.



The pointwise errors are small because the layer is not resolved.

Why, for small fixed N, does the error appear to decrease as ε is reduced?

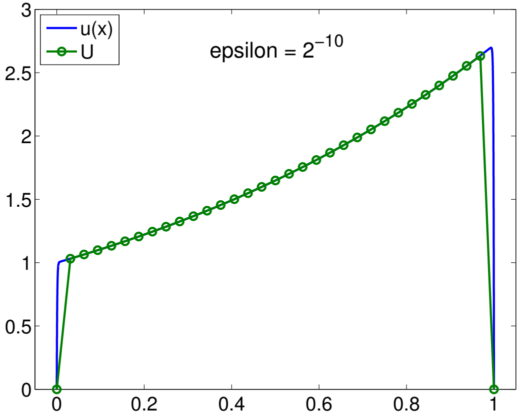
We fix $N = 32$ and take $\varepsilon = 10^{-2}, 10^{-4}, \dots, 10^{-10}$.



The pointwise errors are small because the layer is not resolved.

Why, for small fixed N, does the error appear to decrease as ε is reduced?

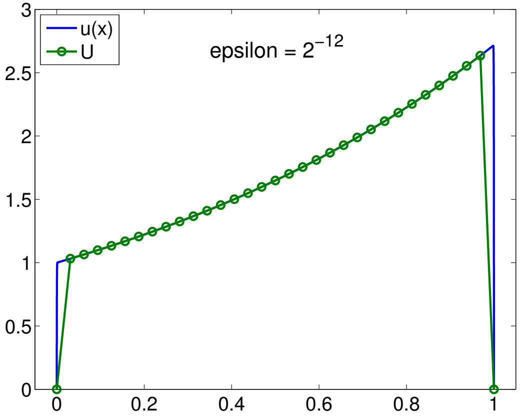
We fix $N = 32$ and take $\varepsilon = 10^{-2}, 10^{-4}, \dots, 10^{-10}$.



The pointwise errors are small because the layer is not resolved.

Why, for small fixed N, does the error appear to decrease as ε is reduced?

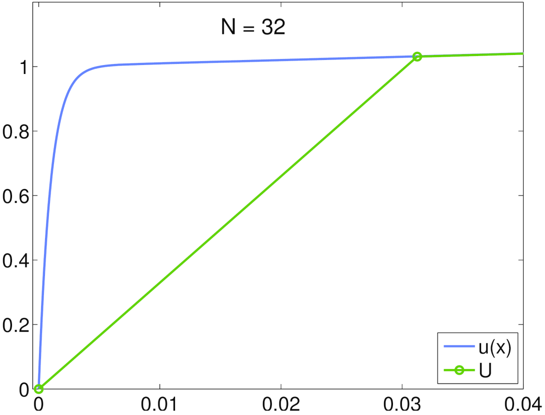
We fix $N = 32$ and take $\varepsilon = 10^{-2}, 10^{-4}, \ldots, 10^{-10}$.



The pointwise errors are small because the layer is not resolved.

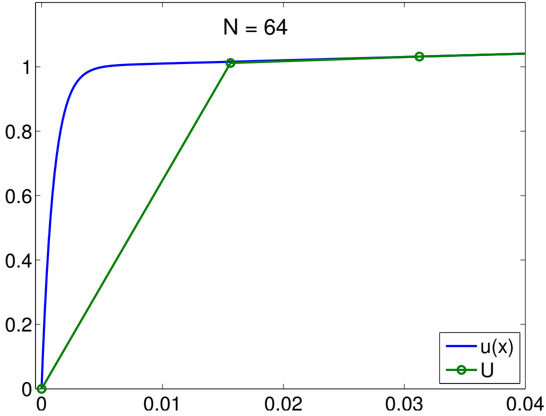
Why, for small fixed ε, does the error appear to increase as N is increased?

We fix $\varepsilon = 2^{-10}$ and take $N = 32, 64, 128, \ldots$. As N approaches ε^{-1}, the method begins to resolve the layer, and so the computed pointwise error increases.



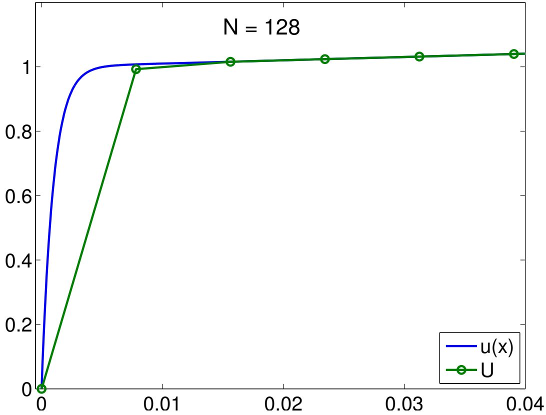
Why, for small fixed ε, does the error appear to increase as N is increased?

We fix $\varepsilon = 2^{-10}$ and take $N = 32, 64, 128, \ldots$. As N approaches ε^{-1}, the method begins to resolve the layer, and so the computed pointwise error increases.



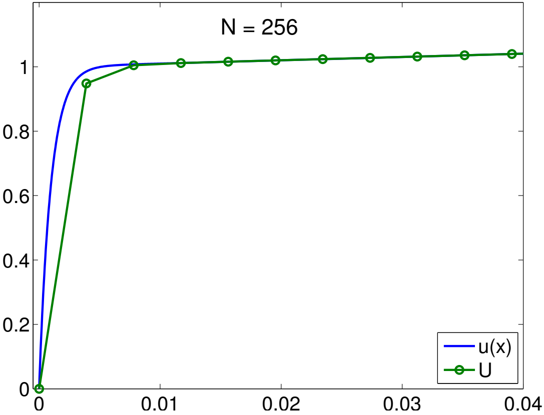
Why, for small fixed ε, does the error appear to increase as N is increased?

We fix $\varepsilon = 2^{-10}$ and take $N = 32, 64, 128, \ldots$. As N approaches ε^{-1}, the method begins to resolve the layer, and so the computed pointwise error increases.



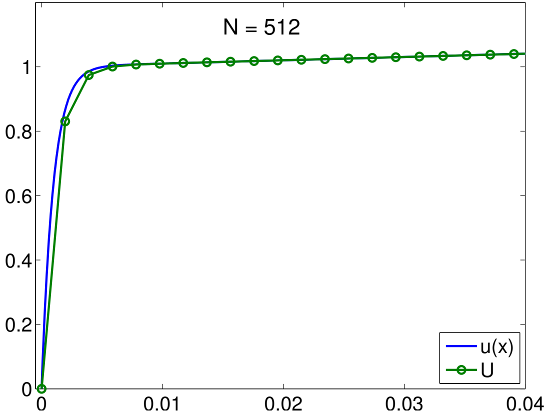
Why, for small fixed ε, does the error appear to increase as N is increased?

We fix $\varepsilon = 2^{-10}$ and take $N = 32, 64, 128, \ldots$. As N approaches ε^{-1}, the method begins to resolve the layer, and so the computed pointwise error increases.



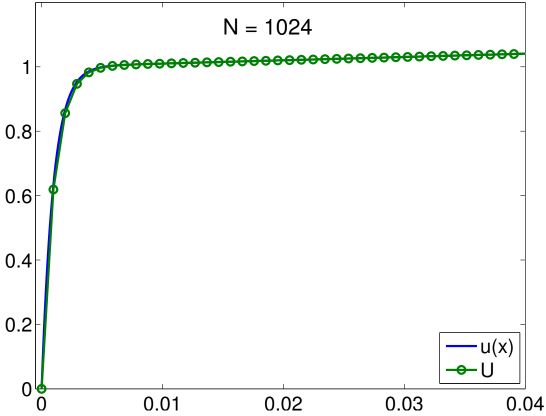
Why, for small fixed ε, does the error appear to increase as N is increased?

We fix $\varepsilon = 2^{-10}$ and take $N = 32, 64, 128, \ldots$. As N approaches ε^{-1}, the method begins to resolve the layer, and so the computed pointwise error increases.



Why, for small fixed ε, does the error appear to increase as N is increased?

We fix $\varepsilon = 2^{-10}$ and take $N = 32, 64, 128, \ldots$. As N approaches ε^{-1}, the method begins to resolve the layer, and so the computed pointwise error increases.



Motivated by the previous graphs, we compute the difference between the true solution and the *piecewise linear interpolant* to the approximation.

$$\max_{0 \leqslant x \leqslant 1} |\mathbf{u}(x) - \bar{\mathbf{U}}(x)|$$

ε^2	$N = 64$	$N = 128$	$N = 256$	$N = 512$
1	3.75e-01	3.75e-01	3.75e-01	3.75e-01
1e-02	3.77e-01	3.75e-01	3.75e-01	3.75e-01
1e-04	4.62e-01	4.06e-01	3.84e-01	3.78e-01
1e-06	7.30e-01	6.86e-01	5.94e-01	4.89e-01
1e-08	7.50e-01	7.49e-01	7.47e-01	7.37e-01
1e-10	7.50e-01	7.50e-01	7.50e-01	7.50e-01
1e-12	7.50e-01	7.50e-01	7.50e-01	7.50e-01

We can conclude from this that the method given here is not suitable for this problem.

Most of the remainder of the next class will be given over to deriving and analysing a method that *is* suitable.

The differential equation is $-\varepsilon^2 u'' + bu = f.$
From this we see that $\|u''\|$ is $\mathcal{O}(\varepsilon^{-2})$.
If b is constant, by differentiating the DE, we get that $\|u^{(4)}\|$ is $\mathcal{O}(\varepsilon^{-4})$. (We will do this more carefully for variable b later).

If standard arguments based on the truncation error are employed, one will find that

$$\|u - U\|_\infty \leqslant CN^{-2}(1 + \varepsilon^{-2}).$$

(This bound suggests that this method is inappropriate for this problem, which is true; but it is not sharp. We will return to this point later).

A convection-diffusion problem

Before we study the reaction-diffusion problem in greater depth, we take a detour to point out that things could be much, much *worse*.

The numerical method presented above yields a reasonable solution to the reaction-diffusion problem *away from layers*.

If we apply the method to the obvious *convection-diffusion problem*, the resulting solution can be unstable.

Again we start with the uniform mesh with N intervals:

$$\Omega^N := \{x_i\}_{i=0}^N, \qquad \text{where } x_i = i/N = ih;$$

And again approximate u'' as $u'' = \delta^2 u(x_i) + K_2 N^{-2}$.

We approximate u' by the corresponding second-order central difference scheme

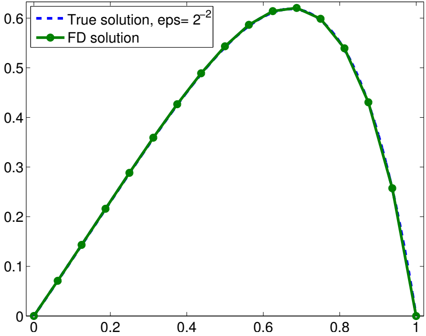
$$u' = \underbrace{\frac{1}{2h}\big(-u(x_{i-1}) + u(x_{i+1})\big)}_{D^0 u(x_i)} + K_1 N^{-2}.$$

The finite difference method is then

$$-\varepsilon\delta^2 U_i + a(x_i) D^0 U_i = f(x_i), \qquad i = 1, \dots, N-1.$$

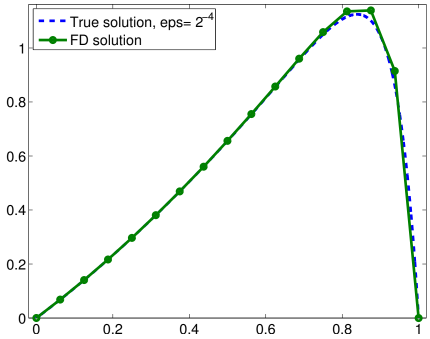
A convection-diffusion problem

$$-\varepsilon u''(x) + u'(x) = 1 + x, \quad \text{on } \Omega = (0,1), \text{ with } u(0)=u(1)=0$$



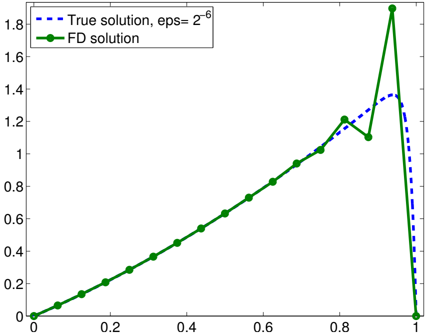
A convection-diffusion problem

$$-\varepsilon u''(x) + u'(x) = 1 + x, \quad \text{on } \Omega = (0,1), \text{ with } u(0)=u(1)=0$$



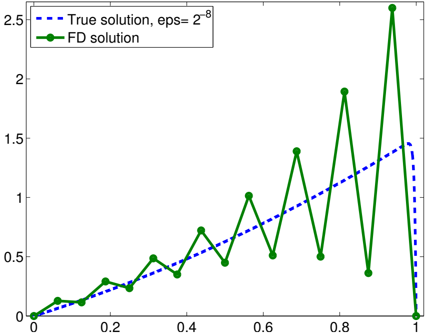
A convection-diffusion problem

$$-\varepsilon u''(x) + u'(x) = 1 + x, \quad \text{on } \Omega = (0,1), \text{ with } u(0)=u(1)=0$$



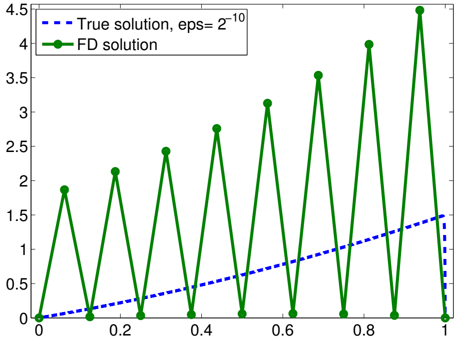
A convection-diffusion problem

$$-\varepsilon u''(x) + u'(x) = 1 + x, \quad \text{on } \Omega = (0,1), \text{ with } u(0)=u(1)=0$$



A convection-diffusion problem

$$-\varepsilon u''(x) + u'(x) = 1 + x, \quad \text{on } \Omega = (0,1), \text{ with } u(0)=u(1)=0$$



Uniform convergence

We have seen that the two simplest methods for reaction-diffusion and convection-diffusion problems are inadequate.

Before constructing a method that *is* adequate, we need a concept of what "adequate" means.

Although it is possible to design a method that gives a "reasonable" solution for a fixed, small ε, we want to investigate schemes which are accurate for *all* $\varepsilon \in (0, 1]$.

Furthermore, the scheme should not rely on choosing some large $N = N(\varepsilon)$ in order to ensure accuracy.

That is...

[Farrell et al., 2000, p10]

... we undertake ... the task of constructing numerical methods that generate numerical solutions which converge uniformly for all values of the parameter ε in the range $(0, 1]$, and that require a parameter-uniform amount of computational work to compute each numerical solution.
Such methods are called parameter uniform or ε-uniform methods.

Uniform convergence

[Farrell et al., 2000, p10]

If a method is ε-uniform, the error between the exact solution, u, and the numerical solution U, satisfies an estimate of the following form: for some positive integer N_0, all integers $N \geqslant N_0$, and all $\varepsilon \in (0,1]$, we have

$$\|u - \bar{U}\|_{\bar{\Omega}} \leqslant CN^{-p}.$$

where C, N_0 and p are positive constants independent of ε and N.

Here U is taken to be a *mesh function* defined on some set of (mesh) points in the domain $\bar{\Omega}$, and $\bar{U}$ is its piecewise linear interpolant. The norm $\|u - \bar{U}\|_{\bar{\Omega}}$ is the maximum norm.

In the above discussion,

- the emphasis on the maximum norm comes from the fact that other norms, particularly energy norms for simple Galerkin FEMs, are not strong enough to identify layers.
- the interpolant of the numerical solution features since, as we have seen, if there are no mesh points within the layer, the solution can appear highly accurate.

So [Farrell et al., 2000] propose that methods for SPPs should be

(1) **global**: yielding an approximation that can be evaluated at all points in the domain;

(2) **point-wise accurate**,

(3) **parameter uniform** (independent of ε and computational effort)

(4) **monotone** (discrete operator respects key qualitative properties of the continuous operator).

References

Farrell, P. A., Hegarty, A. F., Miller, J. J. H., O'Riordan, E., & Shishkin, G. I. (2000).
Robust Computational Techniques for Boundary Layers.
Number 16 in Applied Mathematics. Boca Raton, U.S.A.: Chapman & Hall/CRC.

Linß, T. (2010).
Layer-adapted meshes for reaction-convection-diffusion problems, volume 1985 of *Lecture Notes in Mathematics*.
Berlin: Springer-Verlag.