

Introduction

Galerkin methods are a common class of methods used to approximate ordinary and partial differential equations (ODE/PDE). Galerkin methods rely on the selection of a set of basis functions that are used to represent the solution of the differential equation. Typical basis functions include: piecewise linear and quadratic polynomials and sine/cosine functions for spectral methods

From areas of image compression to speech recognition wavelets have had a profound impact on representing large and small scale datasets in the computational science realm. Wavelets also have a number of features that make them attractive functions to work with including: multi-resolution, compact support, differentiability and orthogonality.

Reduced Order Modeling (ROM) is a widely used method to reduce the computational cost solving differential equations when using standard techniques like the Finite Element Method (FEM). This research will demonstrate the viability of using ROM with the Wavelet Galerkin approach to solving ODE's.

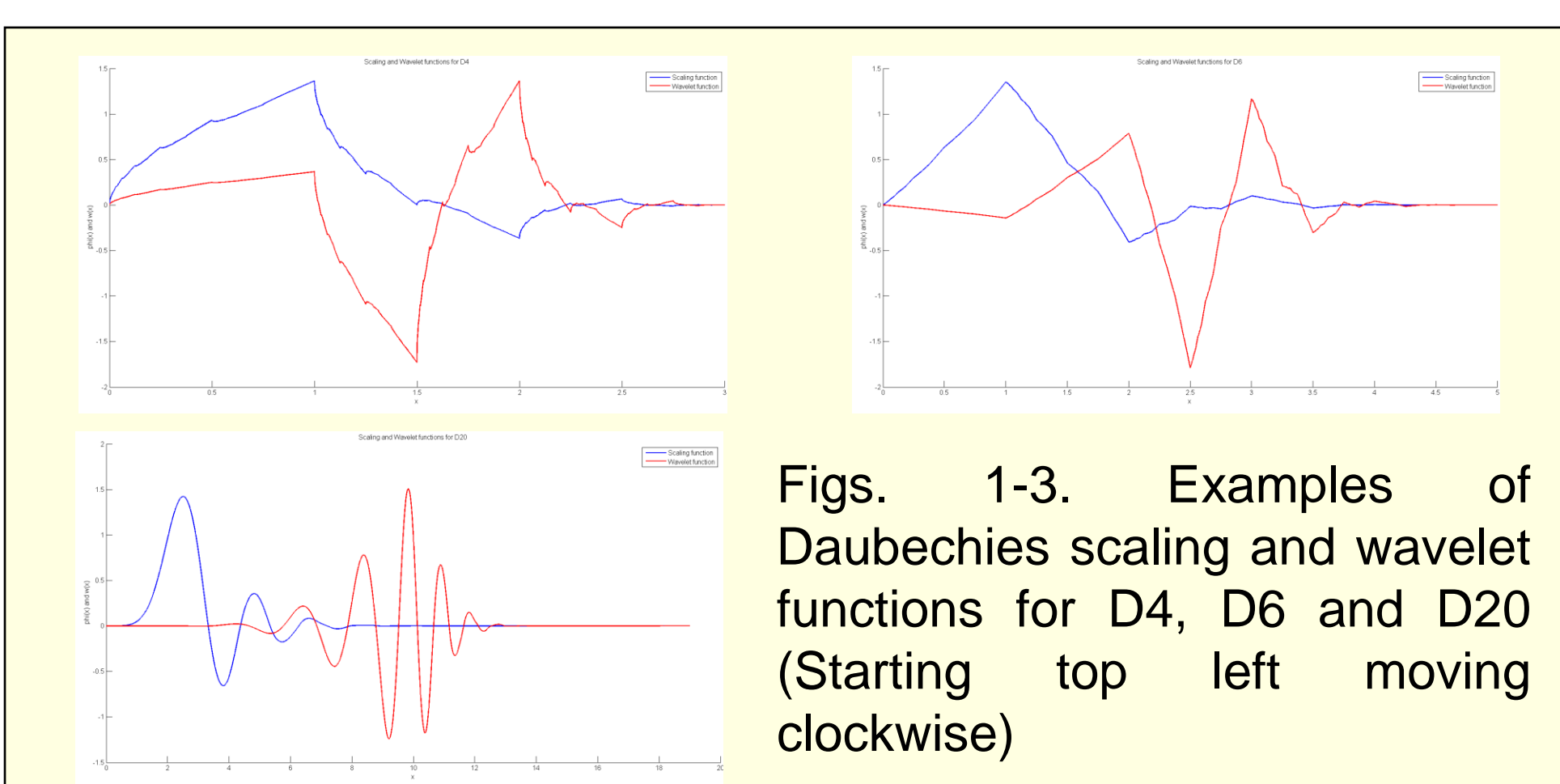
The Wavelet-Galerkin method

The first thing to be done in this process is the selection of our wavelet basis functions. There are many choices available including: Legendre Daubechies orthogonal and biorthogonal wavelets. But to keep things simple for now we will choose the Daubechies family of orthogonal wavelets. Daubechies wavelets are constructed to maximize the number of vanishing moments which is correlated to polynomial order the wavelet can approximate. One advantage of Daubechies scaling functions, which are functions that define a given wavelet, is that they are compactly supported over a given domain. Typically Daubechies wavelets are referred to in terms of their support DN; so the wavelet with support over $[0,3]$ is called D4, $[0,5]$ is called D6 etc.

Unfortunately a problem with wavelets, that doesn't exist in many of the other standard basis functions, is we do not have an explicit formula to calculate the function values. In order to construct the basis function though we can use what is called the dilation equation:

$$\phi(x) = \sum_k a_k \phi(2^m x - k) \quad [1]$$

where a_k are coefficient values determined by the type of wavelet. One can use a recursive method, or what's known as the Cascade algorithm, to approximate the function values on a given domain.



Figs. 1-3. Examples of Daubechies scaling and wavelet functions for D4, D6 and D20 (Starting top left moving clockwise)

Now that our basis function has been chosen we can begin to formulate our ODE. Using homogeneous Dirichlet boundary conditions on $x \in [0,1]$ we seek a discrete u^h satisfying our boundary conditions with the differential equation:

$$-\beta u_{xx} + \gamma u_x + \alpha u = f(x) \quad [2]$$

where u is the solution to our differential equation and β, γ and α are constants. But first lets take a look at the weak form:

$$-\beta \int u_{xx} v dx + \gamma \int u_x v dx + \alpha \int u v dx = \int f(x) v dx \quad [3]$$

We will seek $u^h \in V^m$ where V^m is defined as the space spanned by all levels(m) and translates(k) of our scaling function $\phi(2^m x)$. Since $u^h \in V^m$ and $\phi(2^m x - k)$ form a basis, we can write:

$$u^h = \sum C_{m,k} \phi(2^m x - k) \quad [4]$$

where $C_{m,k}$ will be the unknowns in our weak problem.

The Wavelet-Galerkin method cont.

Using this definition of u^h we can re-write our weak problem. The first term in the problem would look like:

$$\sum C_{m,k} [-\beta \int \phi''(2^m x - l) \phi(2^m x - k) dx] \quad [5]$$

where m determines the spacing between our basis functions, and l and k are the scaling function translates.

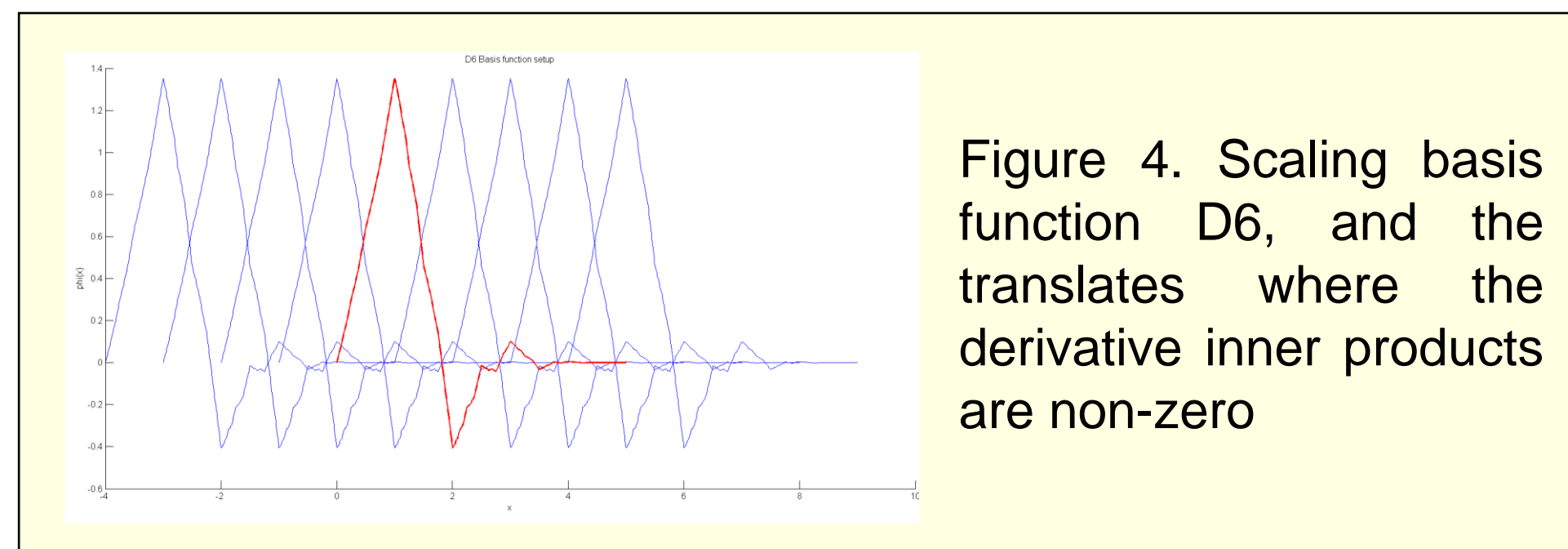


Figure 4. Scaling basis function D6, and the translates where the derivative inner products are non-zero

In order to calculate the inner products of this problem we must use a method proposed by Latto et. al. to find what are called connection coefficients. These connection coefficients represent the inner products between two scaling functions at a given derivative, d . Since the scaling functions are orthogonal, we only need the connection coefficients for the terms with derivatives in them because the non-derivative terms are only non-zero when $k = l$.

Once the connection coefficients have been calculated, all that needs to be done is to resolve the boundary conditions and then set up a system of equations to solve the problem for our $C_{m,k}$'s. There are two typical approaches to resolve the boundary conditions:

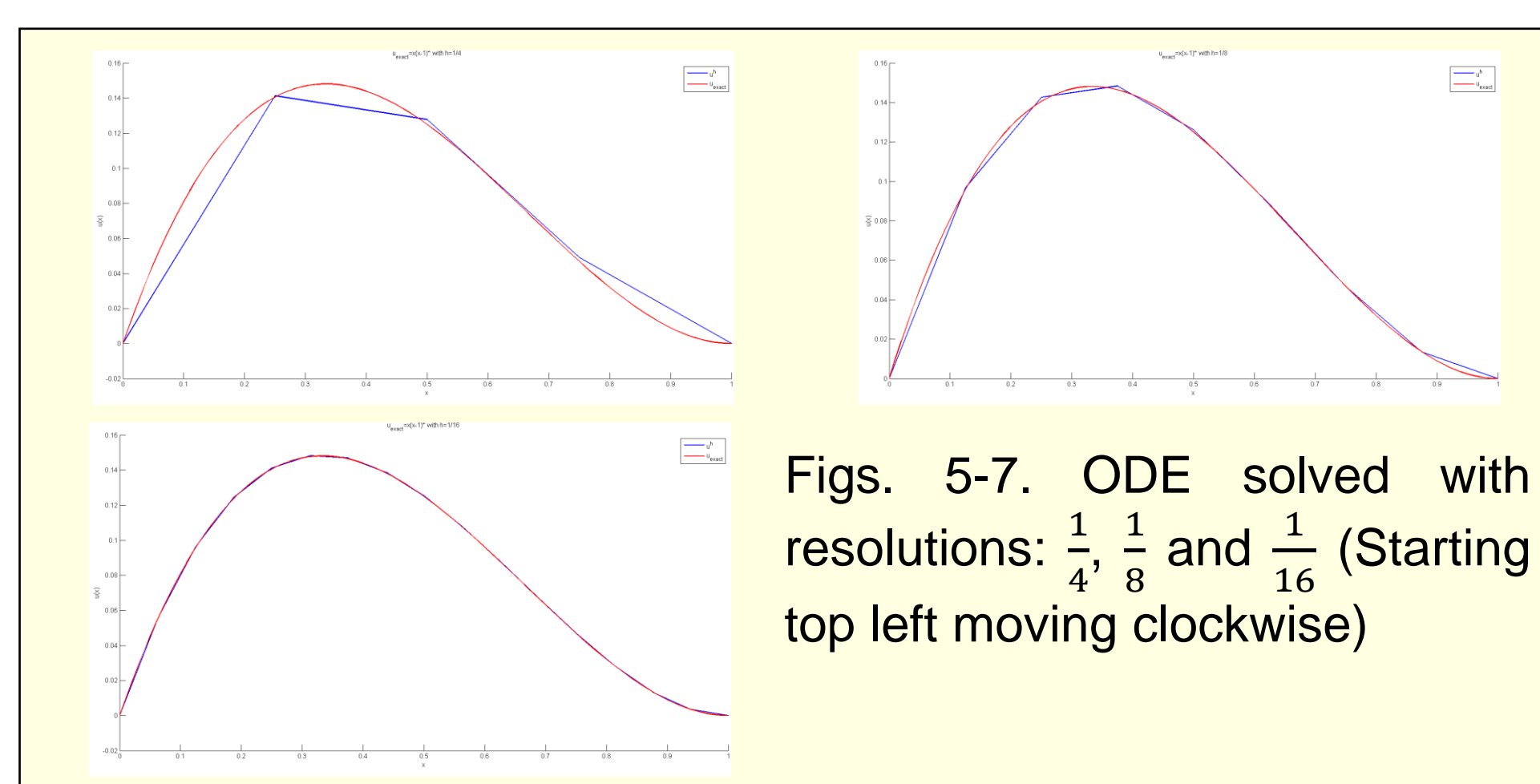
- Add N-1 "phantom" basis functions that extend past the ends of the domain to compute the inner products of our basis functions near the boundaries.
- Modify the connection coefficients near the boundaries.

We will choose to extend our basis functions past the ends of our domain. Doing so leaves us with a sparse banded system of equations comprised of a combination of our connection coefficients for their respective terms.

Now, to ensure that our Wavelet-Galerkin method is working we will formulate a test problem with a known solution that satisfies our homogeneous Dirichlet boundary condition requirement so that we can compute the error and determine the rates of convergence as we increase our discretization. The exact solution we will use is:

$$u_{exact} = x(x-1)^2 \text{ with } x \in [0,1] \quad [6]$$

So if we define our constants as $\beta = \gamma = \alpha = 1$ our right hand side becomes $f(x) = x^3 + x^2 - 9x + 5$; and we have everything we need to solve the ODE.



Figs. 5-7. ODE solved with resolutions: $\frac{1}{4}$, $\frac{1}{8}$ and $\frac{1}{16}$ (Starting top left moving clockwise)

From the plots in figures 5-7 we see that the Wavelet-Galerkin method appears to approach the actual solution as we hoped. Using the exact solution and the computed solutions at a number of discretization's we can calculate the rates of convergence as the resolution is increased. Remember that h is calculated with respect to 2^m so as we increase m , the resolution is also increased

h	Euclidean distance Error	Rate
0.25	0.018618	
0.125	0.0117	0.696796
0.0625	0.003785	1.504585
0.03125	0.000954	1.722775
0.015625	0.000198	1.886827
0.007813	2.85E-05	2.261854

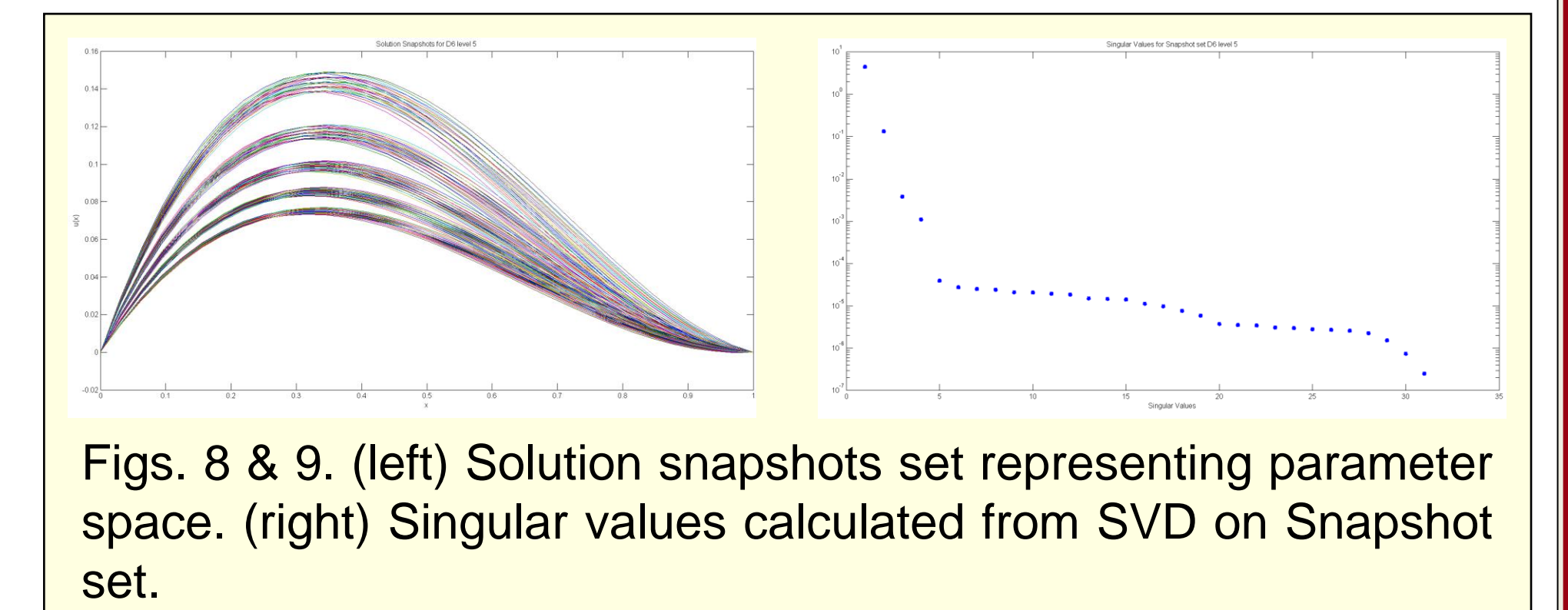
From the rate of convergence table we see that the convergence rates of our Wavelet-Galerkin method do quite well. In fact it approaches and then surpasses a quadratic convergence rate.

Reduced Order Modeling

Now that we have a method of computing solutions to ODE's using the Wavelet-Galerkin method we can begin to format our reduced order model. One way to accomplish this is what's known as the Proper Orthogonal Decomposition (POD). POD uses the Singular Value Decomposition (SVD) to compute an orthogonal set of basis vectors that can be used to construct a solution with only a few degrees of freedom. In SVD the first matrix U , where $A = U\Sigma V^T$, represents the column space of our matrix A .

The first step of creating our reduced order model is a pre-processing step that involves solving the ODE a number of times for a range of parameter values. For our problem we will solve the test problem that was used in the last section while varying β, γ and α between 1 and 2 in $\frac{1}{4}$ increments. This means that we will have 125 solutions to fill our space.

Then we can make what's called a snapshot matrix by compiling the solution column vectors; we will use this as our A matrix in the SVD to find U (our reduced basis vectors).



Figs. 8 & 9. (left) Solution snapshots set representing parameter space. (right) Singular values calculated from SVD on Snapshot set.

The basis functions computed using the SVD will now act as our basis functions to calculate a ROM solution to our ODE. It is important to note that these basis functions are not compactly supported as they exist over our entire domain implying that our matrix will be a dense system. But the hope is that in the end it will take less work to solve a small dense linear system as opposed to a large sparse system.

The final step in this process is to construct a ROM solution as a linear combination of our reduced basis functions ($\psi(x)$), which in turn are linear combinations of our scaling functions, $\phi(2^m x - k)$.

$$u_{ROM} = \sum \mu_j \psi_j(x) \quad [7]$$

where the μ_j 's are our values to be computed.

Using the weak problem and our reduced basis function we can formulate an equation to calculate the ROM solution. The first term is given by:

$$-\beta \sum_j \mu_j \int (\sum_j C_j \phi_j(2^m x - k)) \times (\sum_i C_i \phi_i(2^m x - l)) dx \quad [8]$$

As we can see from this equation we are going to need to compute a number of dot products between the reduced basis functions and our connection coefficients.

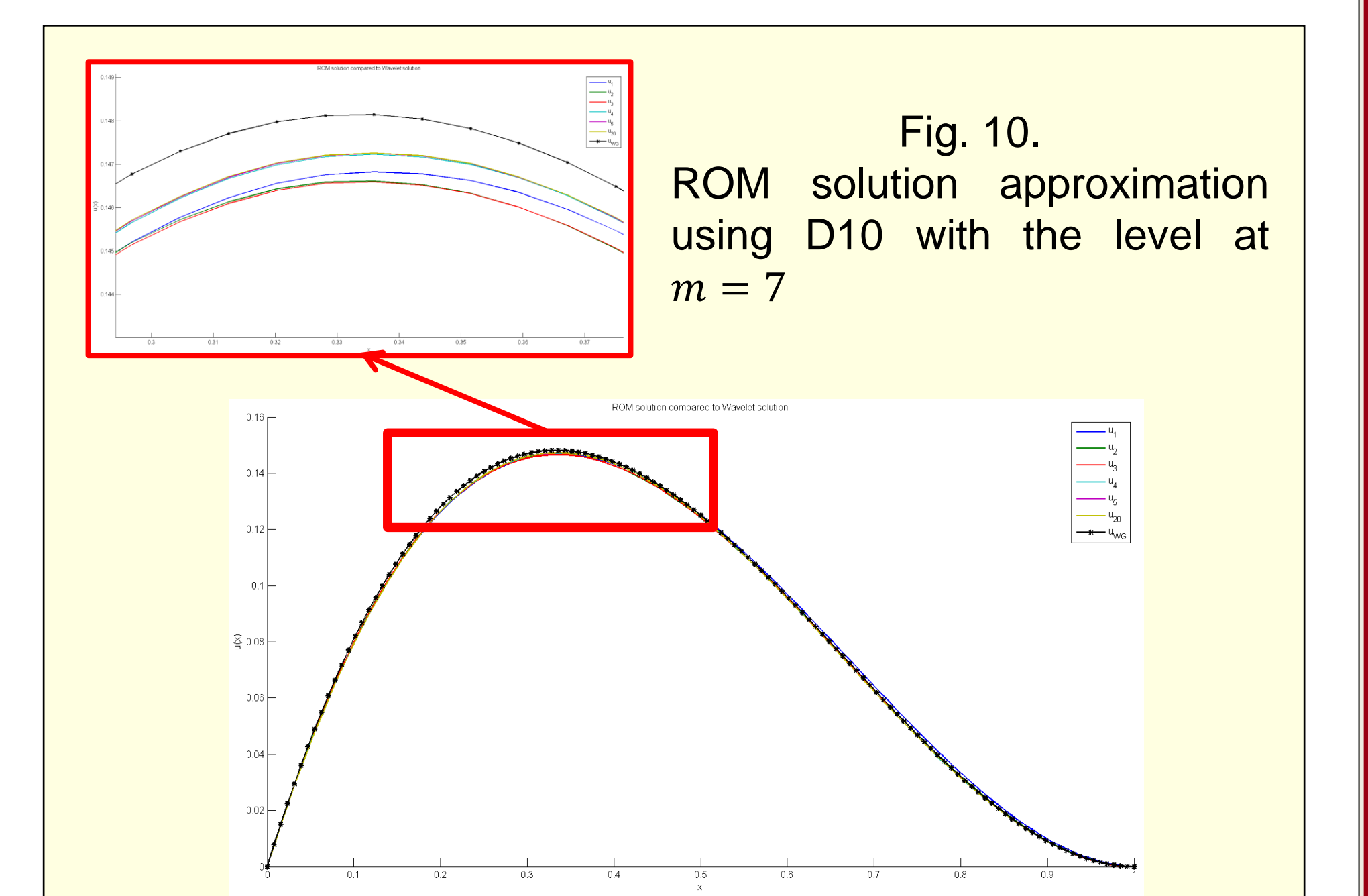


Fig. 10. ROM solution approximation using D10 with the level at $m = 7$

Bibliography and Acknowledgement

1. Besora, Jordi, Galerkin Wavelet Method for Global Waves in 1D, Master Thesis, Royal Inst. of Tech. (Sweden), 2004.
2. A. Latto, H.L. Resnikoff and E. Tenenbaum, The Evaluation of Connection Coefficients of Compactly Supported Wavelets, 1991, SpringerVerlag, 1992.

Finally I would like to thank Max Gunzburger for support of my research this past semester.