Adaptive spectral graph wavelet method for PDEs on network

by

Ankita Shukla

(IIT Delhi)

under the supervision of

Mani Mehra and Günter Leugering

Benasque, 29 August, 2017

- The continuous model
- Adaptive spectral graph wavelet method
- Problem 1
- Problem 2
- Problem 3
- Conclusion

Continuous model

- A weighted graph $G = \{E, V, \omega\}$ consists of
  - a set of edges E,
  - a set of vertices (or nodes) V,
  - a weighted function $\omega : E \to \mathbb{R}^+$ which assigns a positive weight to each edge.
- The graph is finite ,i.e., dim(V)= $N < \infty$.

Continuous model

- The adjacency matrix $A = \{a_{m,n}\}$ for the weighted graph G is the $N \times N$ matrix where

$$a_{m,n} = \begin{cases} \omega(e) & \text{if } e \in E \text{ connects vertices } m \text{ and } n, \\ 0 & \text{otherwise.} \end{cases}$$

- The matrix A is a symmetric matrix.
- Vertices $V = \{v_1, v_2, \cdots, v_N\}$ are divided into two disjoint sets V^B and V^C

$$V^B = \{v_i | v_i \in V; i = 1, \cdots, N; \sum_{k=1}^{s} |a_{i,k}| = 1\},$$

$$V^C = \{v_i | v_i \in V; i = 1, \cdots, N; \sum_{k=1}^{s} |a_{i,k}| > 1\}.$$

- $I(v) = \{k \in \{1, \cdots, s\} | e_k \in E; v \in e_k; v \in V\}$.
  $I(v)$:Set contains all indices of edges connected to the node v.

Continuous model

The spaces for the network domain are

$$\mathcal{H} = \prod_{k=1}^{s} \mathcal{L}_2(\Omega_k), \quad \Omega_k = (0,1)$$

$$V = \{w \in \prod_{k=1}^{s} \mathcal{H}_1(\Omega_k) | w_k(v) = w_l(v); \forall v \in V^C; k, l \in I(v)\}.$$

with the corresponding norms

$$\|w\|_{\mathcal{H}} = \left(\sum_{k=1}^{s} \|w_k\|^2_{\mathcal{L}_2(\Omega_k)} \right)^{\frac{1}{2}} \quad \text{and} \quad \|w\|_V = \left(\sum_{k=1}^{s} \|w_k\|^2_{\mathcal{H}_1(\Omega_k)} \right)^{\frac{1}{2}},$$

where $\mathcal{L}_2(\Omega_k)$ and $\mathcal{H}_1(\Omega_k)$ are standard spaces.

Continuous model

- Consider the following network with linear and parabolic PDE on each edge:

$$\frac{\partial u_k(x,t)}{\partial t} + \mathcal{M}(u_k(x,t)) = f_k(x,t), \quad x \in \Omega_k, t > 0, k = 1, \cdots, s,$$

$$u_k(v_i, t) = g_i, \quad v_i \in V^B, k \in I(v_i), t > 0,$$

$$u_k(x, 0) = u_k^0(x), \quad x \in \Omega_k, k = 1, \cdots, s,$$

$$u_k(v_i, t) = u_l(v_i, t), \quad v_i \in V^C, k \neq l \in I(v_i), t > 0,$$

(Continuity condition)

Continuous model

$$\sum_{k \in I(v_i)} \nu B_{i,k} \frac{\partial u_k(v_i, t)}{\partial x} = 0, \quad v_i \in V^C, t > 0.$$

(Kirchoff condition)

- B:Incidence matrix

$$B_{i,k} = \begin{cases} 1 & \text{if } v_i = v_k^{\text{start}} \\ -1 & \text{if } v_i = v_k^{\text{end}} \\ 0 & \text{else} \end{cases}$$

$$\mathcal{M}(u_k(x,t)) = c\frac{\partial u_k(x,t)}{\partial x} - \nu\frac{\partial^2 u_k(x,t)}{\partial x^2},$$

- ν and c are the constant coefficient of the differential operators.

Star shaped network

- Consider the graph arising from a star shaped simple network

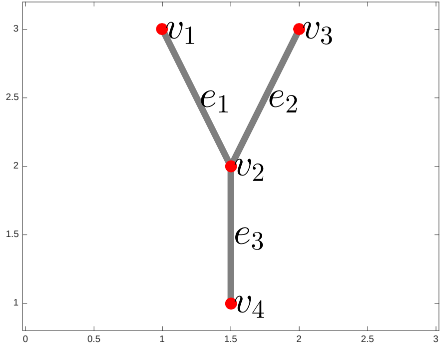


Figure: The star shaped simple network.

Star shaped network

- To solve differential equations on this network, discretization of the graph is needed.

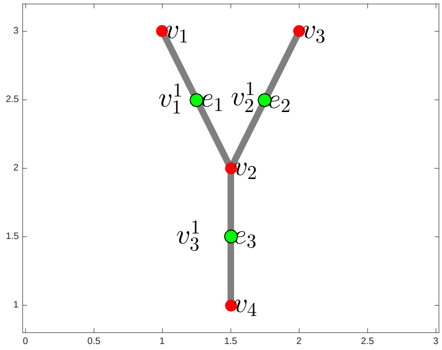


Figure: Discretization of star shaped simple network.

Star shaped network

The adjacency matrix A and the matrix D corresponds to above graph are

$$A = \begin{bmatrix} 0 & \frac{1}{\delta x^2} & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{\delta x^2} & 0 & \frac{1}{\delta x^2} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{\delta x^2} & 0 & \frac{1}{\delta x^2} & 0 & \frac{1}{\delta x^2} & 0 \\ 0 & 0 & \frac{1}{\delta x^2} & 0 & \frac{1}{\delta x^2} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\delta x^2} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\delta x^2} & 0 & 0 & 0 & \frac{1}{\delta x^2} \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{\delta x^2} & 0 \end{bmatrix},$$

$$D = \begin{bmatrix} \frac{1}{\delta x^2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{2}{\delta x^2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{3}{\delta x^2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{2}{\delta x^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\delta x^2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{2}{\delta x^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\delta x^2} \end{bmatrix}.$$

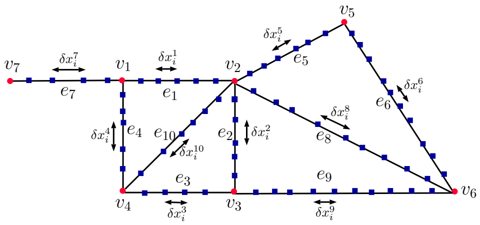
Star shaped network

The approximation of graph Laplacian $\mathcal{L}$ for the whole graph using $L_{\delta x} = D - A$ is given by

$$L_{\delta x} = \frac{1}{\delta x^2} \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 3 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix} .$$

Adaptive spectral graph wavelet method

- Non uniform grid on general network



- Second order accurate central difference approximation for first and second derivative is

$$\left.\frac{\partial u_k}{\partial x}\right|_{x=x_i^k} = \frac{u_{k,i+1} - u_{k,i-1}}{x_{i+1}^k - x_{i-1}^k} + \frac{1}{2}\left.\frac{\partial^2 u_k}{\partial x^2}\right|_{x=x_i^k}(\delta_{x_i^k} - \delta_{x_{i-1}^k}) + O(\delta x^k)^2,$$

Adaptive spectral graph wavelet method

$$\left.\frac{\partial^2 u_k}{\partial x^2}\right|_{x=x_i^k} = \frac{2u_{k,i+1}}{\delta_{x_i^k}(\delta_{x_i^k}+\delta_{x_{i-1}^k})} - \frac{2u_{k,i}}{\delta_{x_i^k}\delta_{x_{i-1}^k}} + \frac{2u_{k,i-1}}{\delta_{x_{i-1}^k}(\delta_{x_{i-1}^k}+\delta_{x_i^k})} + \frac{1}{3}\left.\frac{\partial^3 u_k}{\partial x^3}\right|_{x=x_i^k}(\delta_{x_i^k}-\delta_{x_{i-1}^k}) + O(\delta x^k)^2,$$

- Using above two equations and Crank nicolson scheme for time discretization, we get following first order accurate difference scheme at (x_i^k, t_n) for $i = 1, \cdots, N^k - 1$, $k = 1, 2, \cdots, s$.

$$\frac{U_{k,i}^{n+1} - U_{k,i}^n}{\delta t} = \alpha_i^k(U_{k,i+1}^n + U_{k,i+1}^{n+1}) + \beta_i^k(U_{k,i}^n + U_{k,i}^{n+1})$$

$$+\gamma_i^k(U_{k,i-1}^n + U_{k,i-1}^{n+1}),$$

Adaptive spectral graph wavelet method

- where $U^n_{k,i} \approx u_k(x^k_i, t_n)$,
  $\alpha^k_i = \frac{2\nu}{\delta_{x^k_i}(\delta_{x^k_i}+\delta_{x^k_{i-1}})} + \frac{c}{x^k_{i+1}-x^k_{i-1}}$, $\beta^k_i = -\frac{2\nu}{\delta_{x^k_i}\delta_{x^k_{i-1}}}$ and
  $\gamma^k_i = \frac{2\nu}{\delta_{x^k_{i-1}}(\delta_{x^k_i}+\delta_{x^k_{i-1}})} - \frac{c}{x^k_{i+1}-x^k_{i-1}}$.
- At coupling node (x_0, t_n), we get following first order accurate difference scheme

$$\frac{U^{n+1}_{v_2} - U^n_{v_2}}{\delta t} = -\beta_{v_2}\frac{U^n_{v_2} + U^{n+1}_{v_2}}{2} + \sum_{l=0}^{d_{v_2}} \alpha^l_1 \frac{(U^n_{l,1} + U^{n+1}_{l,1})}{2}.$$

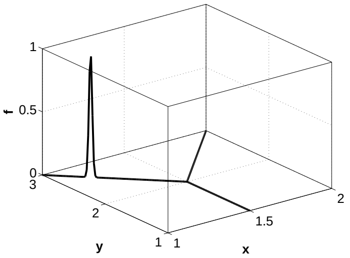
- where $\beta_{v_2} = -\frac{2a}{\sum_{l=0}^{d_{v_2}} \delta_{x_0}{}^2 a_l}$ and $\alpha^l_1 = \frac{2a_l}{\sum_{l=0}^{d_{v_2}} \delta_{x_0}{}^2 a_l}$.

Test function 1

- Consider the test function

$$f = \begin{cases} e^{\frac{-(x-b)^2}{c^2}} & \text{on edge } e_1, \\ 0 & \text{on edge } e_2 \text{ or } e_3. \end{cases}$$

with $b = \frac{1}{3}$ and $c = \frac{1}{64}$



- Observe the behavior of wavelet coefficients and adaptive node arrangement.

Wavelet coefficients

- Important property of wavelet: wavelet coefficients d_k^j decreases rapidly for smooth functions.
- This property makes it suitable to detect where the shocks are located in the numerical solution of a PDE.
- Hence an adaptive node arrangement can be generated.

Wavelet coefficients

- d_k^j are following the region of sharp transition/localized pattern with respect to increasing j.

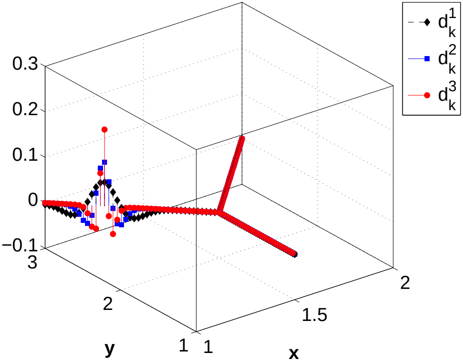


Figure: Wavelet coefficients

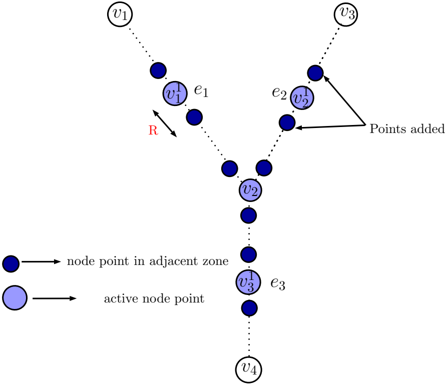
Adaptive node arrangement

x^{new}= **AdaptiveNodeArrangement**(f, x^{old})

- Choose a threshold parameter ϵ and positive adjacent zone constants R and M $(\|(x - x_k)\|_{\mathcal{H}} \leq R)$.
- $m = 0$, $x^m = x^{old}$, $f^m = f$.
- **do while** $m = 0$ or $x^m \neq x^{m-1}$
  - Perform SGWT on f^m to get $\{d_k^J\}_{k=1}^{|x^m|}$.
  - $x^{m+1} = x^m$.
  - Analyse wavelet coefficients $\{d_k^J\}_{k=1}^{|x^m|}$ and collect all the active node points.
  - Corresponding to each active node point, add an adjacent zone in x^{m+1}.
  - Interpolate f^m onto new grid x^{m+1} and call it f^{m+1}.
  - $m = m + 1$
- $x^{new} = x^m$

Adaptive node arrangement

- If $|d_k^j| \geq \epsilon$, x_k is called an active node point.



Test function 1

- The adaptive node arrangement at $\epsilon = 10^{-4}$ for test function is

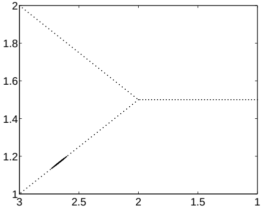


Figure: $N(\epsilon) = 992$.

$N(\epsilon)$: The number of the node points required for $f_{\geq \epsilon}$

- The convergence of graph laplacian on an adaptive grid is $\|\mathcal{L}f - \mathcal{L}_{\delta x} f\|_2 = O(10^{-4})$ at $\epsilon = 10^{-4}$.

Advection–diffusion equation

Consider the advection–diffusion equation on each edge of star graph network

$$\frac{\partial u_k}{\partial t} = c\frac{\partial u_k}{\partial x} + \nu\frac{\partial^2 u_k}{\partial x^2}, \quad x \in \Omega_k, t > 0,$$

$$u_1(v_2, t) = u_2(v_2, t) = u_3(v_2, t) = 0,$$

$$u_1(x, 0) = e^{\frac{-(x-b)^2}{c^2}} \text{with } b = \frac{1}{3}, c = \frac{1}{64},$$

$$u_2(x, 0) = u_3(x, 0) = 0, \quad x \in \Omega_k, k = 2, 3,$$

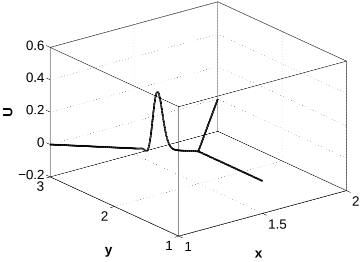
$$u_k(v_i, t) = u_l(v_i, t), v_i \in V^C, k \neq l \in I(v_i), t > 0, k = 1, 2, 3,$$

(Continuity condition)

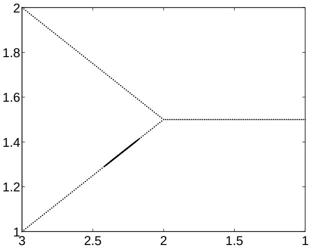
$$\sum_{k\in I(v_i)} \nu B_{i,k} u_k(v_i, t) = 0, v_i \in V^C. \quad \text{(Kirchoff condition)}$$

Advection-diffusion equation

- The solution and corresponding adaptive grid at $t = 0.4$



(a) Solution at $t = 0.4$.

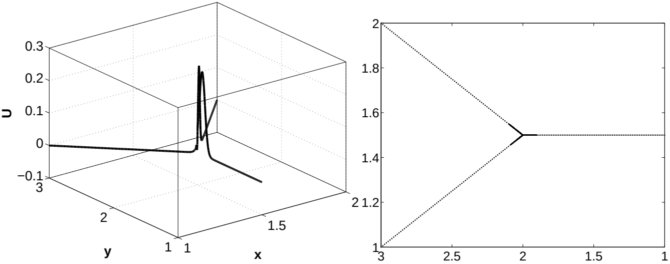


(b) Adaptive node arrangement ($N(\epsilon) = 1870$).

- $c = -1, \quad \nu = 10^{-3}$.

Advection-diffusion equation

- The solution and corresponding adaptive grid at $t = 0.75$

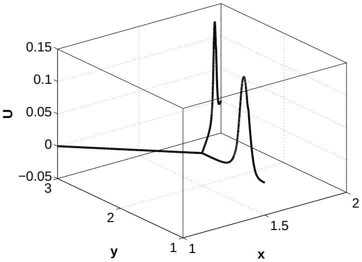


(c) Solution at $t = 0.75$.

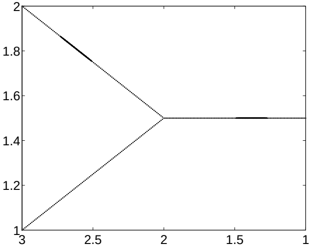
(d) Adaptive node arrangement ($N(\epsilon) = 5192$).

Advection-diffusion equation

- The solution and corresponding adaptive grid at $t = 1.4$



(e) Solution at $t = 1.4$.

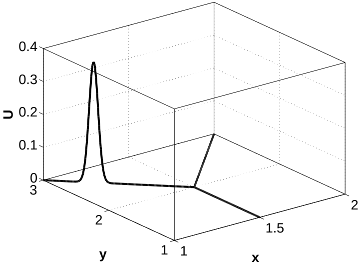


(f) Adaptive node arrangement ($N(\epsilon) = 4164$).

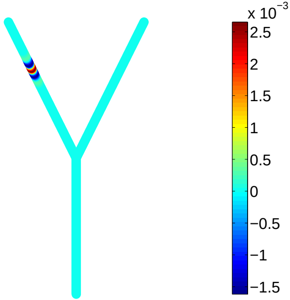
- Sufficiently accurate solution capturing the emergence of the localized patterns at all the scales (including the junction of the network).

Problem 2

- For problem 2, we again take advection–diffusion equation for $c = 0$ which corresponds to heat equation.
- The solution and pointwise error at $t = 0.4$ is



(g) Numerical solution at $t = 0.4$



(h) Pointwise error

Problem 2

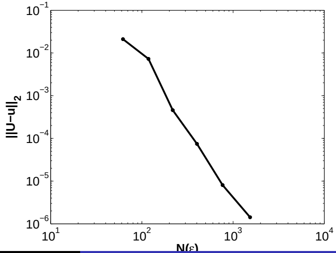
- The analytic solution of heat equation is given by

$$u_1(x,t) = \sum_n A_n \sin(\lambda x) e^{-\nu^2 \lambda^2 t},$$

$$u_2(x,t) = u_3(x,t) = \sum_n B_n(\cos\lambda - \frac{\cos\lambda}{\sin\lambda}) \sin(\lambda x) e^{-\nu^2 \lambda^2 t},$$

where $\lambda = (2n+1)\frac{\pi}{2}$, $n = 0, 1, \cdots$.

- The plot of error versus $N(\epsilon)$



Optimal value of ϵ

- Efficiency constant

$$E = \frac{CPU_{(\epsilon=0)}}{CPU_{(\epsilon)}}$$

- E increases with increase in ϵ.
- But for large value of ϵ, the error $\|U - u\|_p$ is also large.
- Choose an optimal value of ϵ which makes a balance between the efficiency and accuracy.

Problem 3

- Consider the second topology of the network

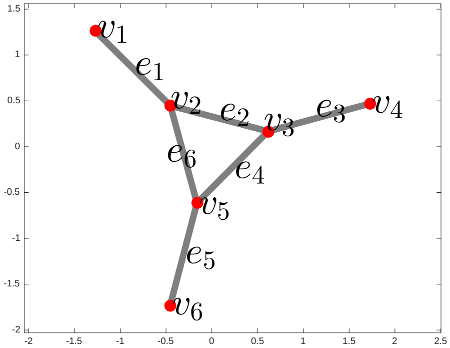
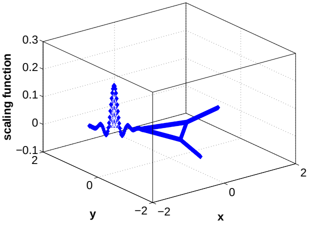


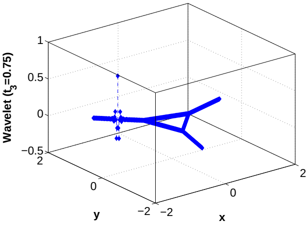
Figure: Topology of the network.

- The scaling function and wavelet at scale $t_3 = 0.75$

Problem 3



(a) The scaling function



(b) The wavelet function at $t_3 = 0.75$

- Second test function on the above topology of the network

$$f = \begin{cases} e^{\frac{-(x-b)^2}{c^2}} & \text{on edge } e_1, \\ 0 & \text{on edge } e_k, k = 2, 3, 4, 5. \end{cases}$$

for $b = \frac{1}{3}, c = \frac{1}{64}$.

Wavelet coefficients

- Any f defined on the vertices of the graph

$$f(n) = \sum_{k=1}^{N} c_k \phi_k(n) + \sum_{j=1}^{J} \sum_{k=1}^{N} d_k^j \psi_k^j(n),$$

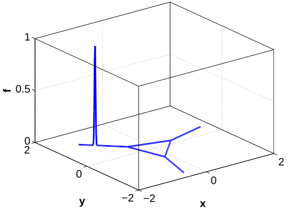
- Given threshold ϵ, above equation can be written as $f(n) = f_{\geq \epsilon}(n) + f_{<\epsilon}(n)$, where

$$f_{\geq \epsilon}(n) = \sum_{k=1}^{N} c_k \phi_k(n) + \sum_{j=1}^{J} \sum_{|d_k^j| \geq \epsilon} d_k^j \psi_k^j(n),$$

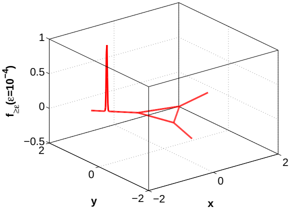
$$f_{<\epsilon}(n) = \sum_{j=1}^{J} \sum_{|d_k^j| < \epsilon} d_k^j \psi_k^j(n).$$

Test function 2

- Test function and corresponding reconstructed function.



(c) Test function 2



(d) $f_{\geq\epsilon}(\epsilon = 10^{-4})$

- $\|f - f_{\geq\epsilon}\|_2$: compression error.

Test function 2

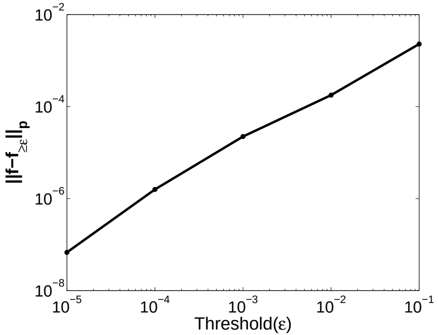


Figure: Variation of compression error versus ϵ.

- Compression error increases with increase in ϵ.

Problem 3

- Heat equation on topology of the network

$$\frac{\partial u_k}{\partial t} = \nu \frac{\partial^2 u_k}{\partial x^2}, \quad x \in \Omega_k, t > 0,$$

$$u_1(v_1, t) = 0, u_3(v_4, t) = 0, u_5(v_6, t) = 0$$

$$u_1(x, 0) = e^{\frac{-(x-b)^2}{c^2}} \text{with } b = \frac{1}{3}, c = \frac{1}{64},$$

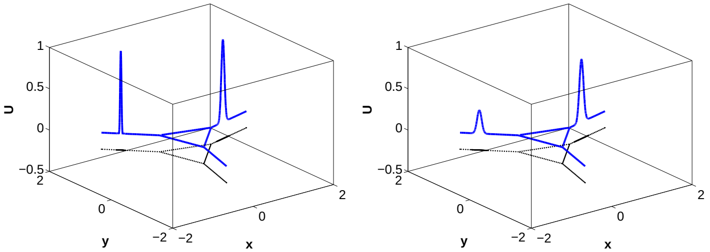
$$u_3(x, 0) = e^{\frac{-(x-b)^2}{c^2}} \text{with } b = \frac{1}{3}, c = \frac{1}{16},$$

$$u_2(x,0) = u_4(x,0) = u_5(x,0) = u_6(x,0) = 0, \quad x \in \Omega_k, k = 2,4,5,$$

$$u_k(v_i, t) = u_l(v_i, t), v_i \in V^C, k \neq l \in I(v_i), t > 0, k = 1, \cdots, 6,$$

$$\sum_{k \in I(v_i)} \nu B_{i,k} u_k(v_i, t) = 0, v_i \in V^C.$$

Problem 3



(a) Initial condition ($N(\epsilon) = 1390$)

(b) Solution ($N(\epsilon) = 7568$)

Figure: Initial condition and solution at $t = 0.7$ and corresponding adaptive grid ($\epsilon = 10^{-4}$).

Conclusion and future work

- ASGWM is used to solve PDEs on two different type of network topology.
- The method uses spectral graph wavelet for the adaptation of the node arrangement
- Same operator is used for the approximation of differential operators and for the construction of spectral graph wavelet.
- We will use ASGWM to solve networks of nonlinear PDEs (e.g Burgers equation) to predict shock waves on complex network.
- Problems based on two and three dimensional networks.

Thank You For Your Attention!