Better Neural PDE Solvers Through Data-Free Mesh Movers

Peiyan Hu[1], Yue Wang[2], Zhiming Ma[1]

1 Academy of Mathematics and Systems Science, Chinese Academy of Sciences

2 Microsoft Research AI4Science

Corresponding to:

hupeiyan18@mails.ucas.ac.cn or

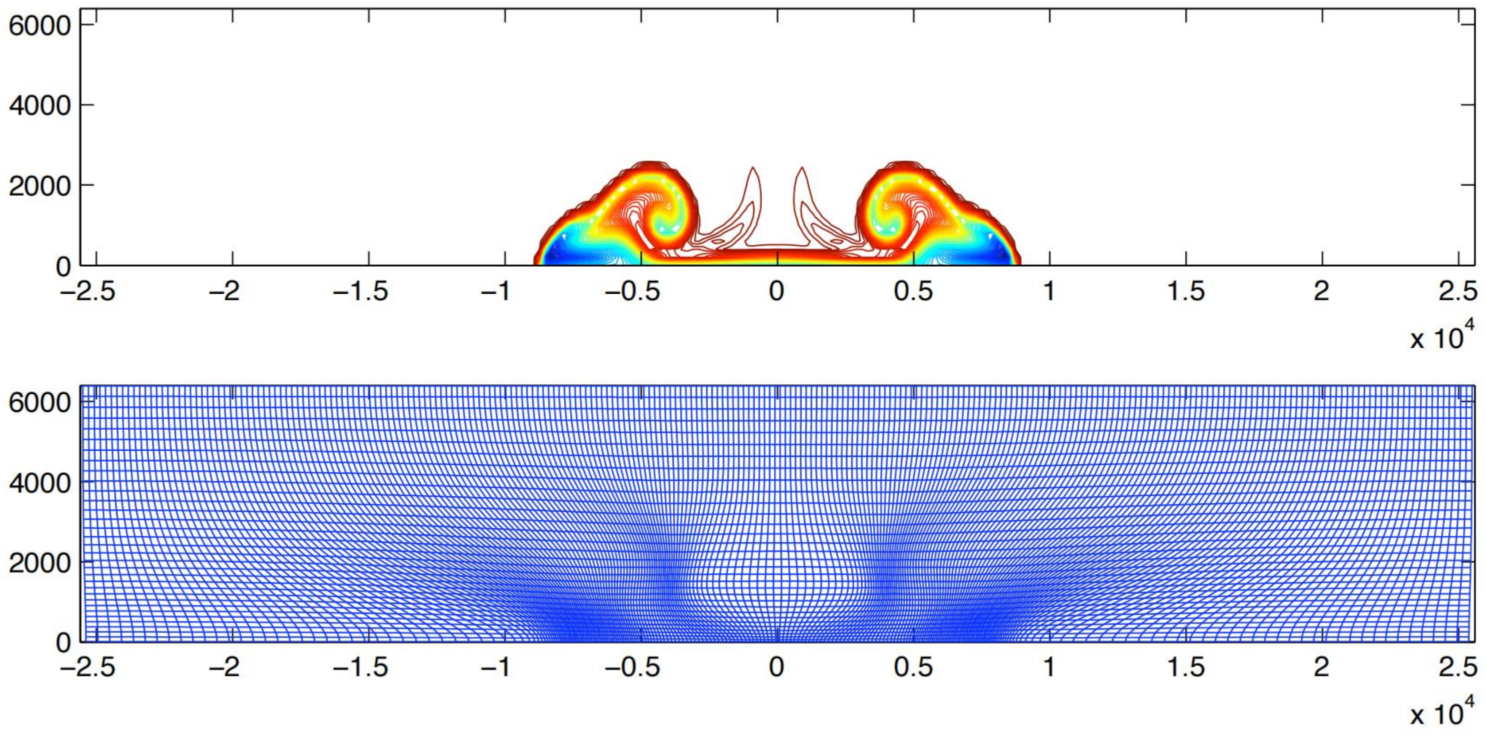
yuwang5@microsoft.com

Outline

- Motivation
- Background
- Theoretic analysis
- Our method:
  - Data-free mesh mover (DMM)
  - Moving mesh based neural PDE solver (MM-PDE)
- Experiment

Motivation

- 1. Predefined static mesh discretizations
- 2. Adaptive and dynamic meshes:
  - (1) Need for expensive optimal mesh data
  - (2) Change of the solution space's degree of freedom and topology during mesh refinement

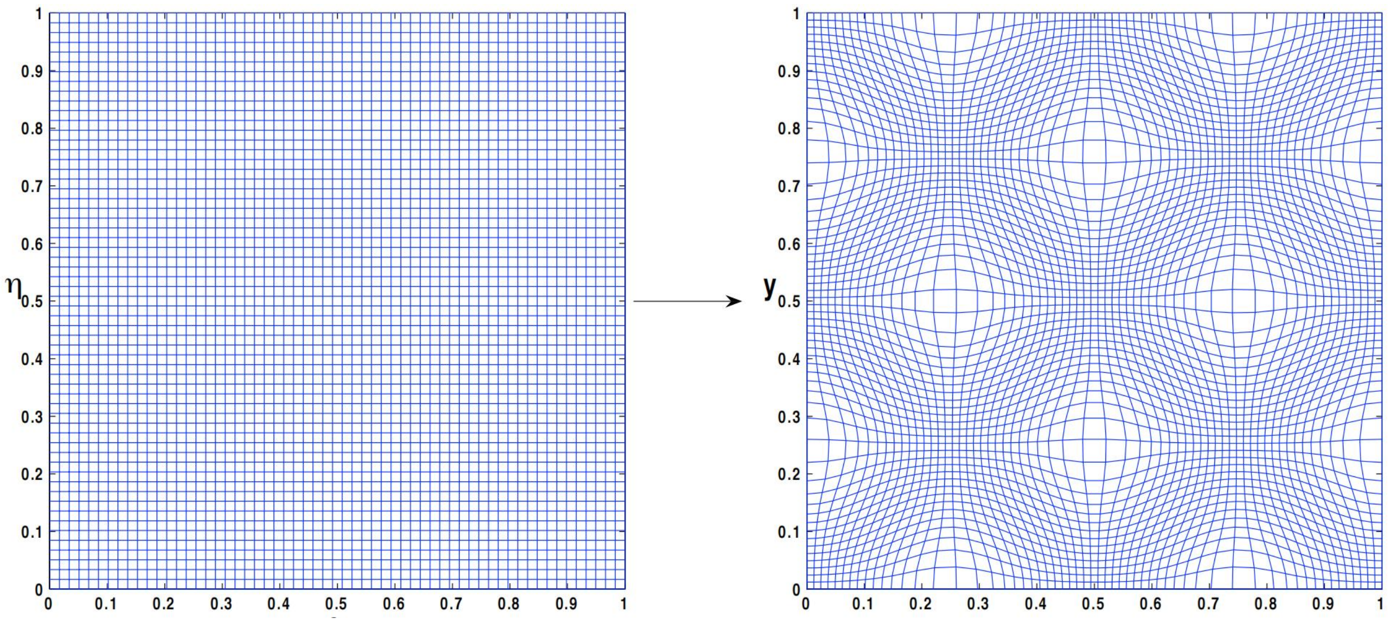


Background

- What is the mesh mover?

  find a target mesh <-> find a coordinate transformation mapping

$$f : [0, T] \times \Omega \rightarrow \Omega,$$

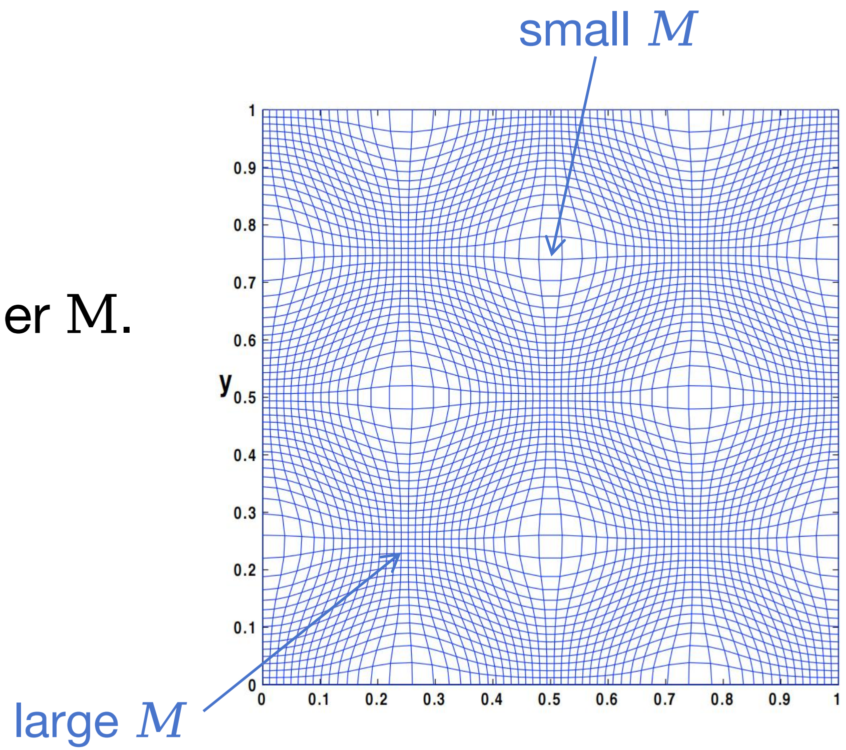


x $\qquad\qquad$ $f(x)$

Background

➢How to define a optimal moving mesh?

- -> Define a function to control the density of mesh: the **monitor function** $M = M(x)$ is a matrix-valued metric function on Ω, its corresponding **mesh density function** $\rho = \sqrt{det(M(x))}$
- ***M*-uniform mesh**: The adaptive mesh moved according to M (or ρ).



- Intuitively, mesh movement is to uniformize cells' volumes under M.
- Region with larger value of M <-> larger mesh density.

Background

- The specific relationship between $M(or\ \rho)$ and the mesh (f):

- **Monge-Ampere equation** (a unique convex solution)

$$\det\left(H(\phi)\right)=\frac{\sigma}{|\Omega|\,\rho\left(\nabla_x\phi\right)},$$

where ϕ is the potential function of f, i.e., $f(x)=\nabla_x\phi$, $H(\phi)$ is the Hessian matrix.

- Boundary condition:

$$\nabla_x\phi(\partial\Omega)=\partial\Omega$$

Choose optimal monitor function with theoretic guarantee

- The form of optimal monitor funciton:
  - The optimal monitor function is proportional to $\langle u \rangle_{W^{l,p}(K)}$.
  - Regions where derivatives are larger -> larger M -> higher mesh densities.

Proposition 1(Huang & Russell, 2010) On mesh $\mathscr{T}$'s every cell K, the optimal monitor function corresponding to the lowest error of interpolation from $W^{l,p}(\Omega)$ to $P_k(\Omega)$ is of the form

$$M_K \equiv \left(1 + \alpha^{-1} \langle u \rangle_{W^{l,p}(K)}\right)^{\frac{2q}{d+q(l-m)}} I, \quad \forall K \in \mathscr{T}, \tag{4}$$

$$\rho_K \equiv \left(1 + \alpha^{-1} \langle u \rangle_{W^{l,p}(K)}\right)^{\frac{dq}{d+q(l-m)}}, \quad \forall K \in \mathscr{T},$$

where $|\cdot|_{W^{m,p}(K)} = \left(\sum_{|\alpha|=m} \int_K |D^\alpha u|^p dx\right)^{\frac{1}{p}}$ is the semi-norm of the Sobolev space $W^{m,p}(K)$, and $\langle \cdot \rangle_{W^{m,p}(K)} \equiv (1/|K|)^{1/p} |\cdot|_{W^{m,p}(K)}$ is the scaled semi-norm.

Choose optimal monitor function with theoretic guarantee

➢ Interpolation error:

Theorem 1 Suppose the error of approximating ρ satisfies

$$||\rho - \tilde{\rho}||_{L_{\frac{d}{d+q(l-m)}}(\Omega)} < \epsilon.$$

Also suppose $u \in W^{l+1,p}(\mathbf{\Omega})$, $q \leq p$ and the monitor function is chosen as in Proposition 1. Let Π_k be the interpolation operator from $W^{l,p}(K)$ to $P_k(K)$. For notational simplicity, we let

$$B(K) = \left(\sum_K |K| \langle u \rangle_{W^{l,p}(K)}^{\frac{dq}{d+q(l-m)}} \right)^{\frac{d+q(l-m)}{dq}}.$$

√ (a) Choose α such that $\alpha \to 0$, if there exists a constant β such that $\|D^l u(\mathbf{x})\|_{l^p} \geq \beta > 0$ a.e. in Ω holds to ensure that M is positive definite, then

$$|u - \Pi_k u|_{W^{m,q}(\Omega)} \leq CN^{-\frac{(l-m)}{d}} B(K).$$

(b) Choose $\alpha \equiv |\Omega|^{-\frac{d+q(l-m)}{dq}} B(K)$, then

$$|u - \Pi_k u|_{W^{m,q}(\Omega)} \leq CN^{-\frac{(l-m)}{d}} B(K)(1 + \epsilon^{\frac{d+q(l-m)}{dq}}).$$

In both cases, as the number of cells N increases to infinity, the right-hand side of the inequality has the bound only relative to the semi-norm of u as

$$\lim_{N \to \infty} B(K) \leq C|u|_{W^{l,\frac{dq}{d+q(l-m)}}(\Omega)}.$$

Our method

- Data-free Mesh Mover (DMM):
  - without optimal mesh data
  - changes the mesh by moving existing nodes rather than adding or deleting nodes and edges

↓ based on DMM

- MM-PDE:
  - a two-branch architecture
  - a learnable interpolation framework

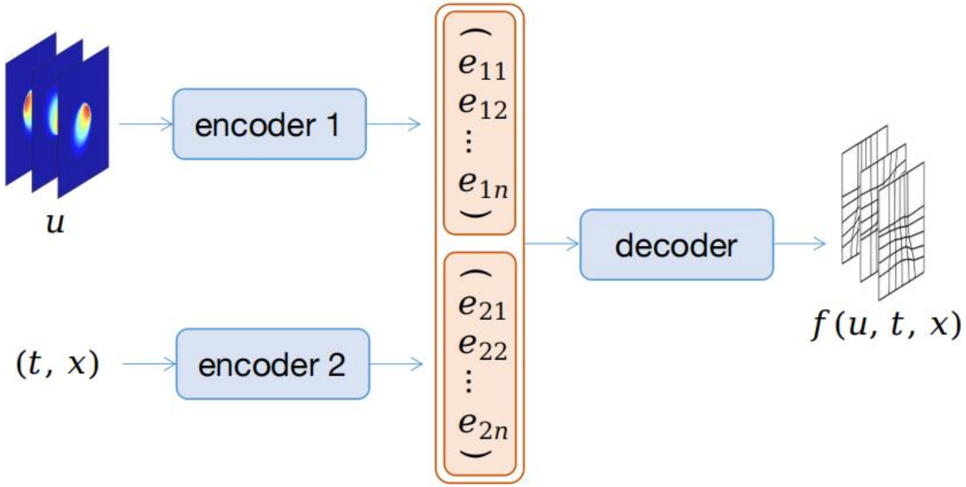
Data-free Mesh Mover (DMM)

- We let the output of the model approximate the residual ψ

$$\nabla_x \psi = \nabla_x \phi - x = f(x) - x.$$

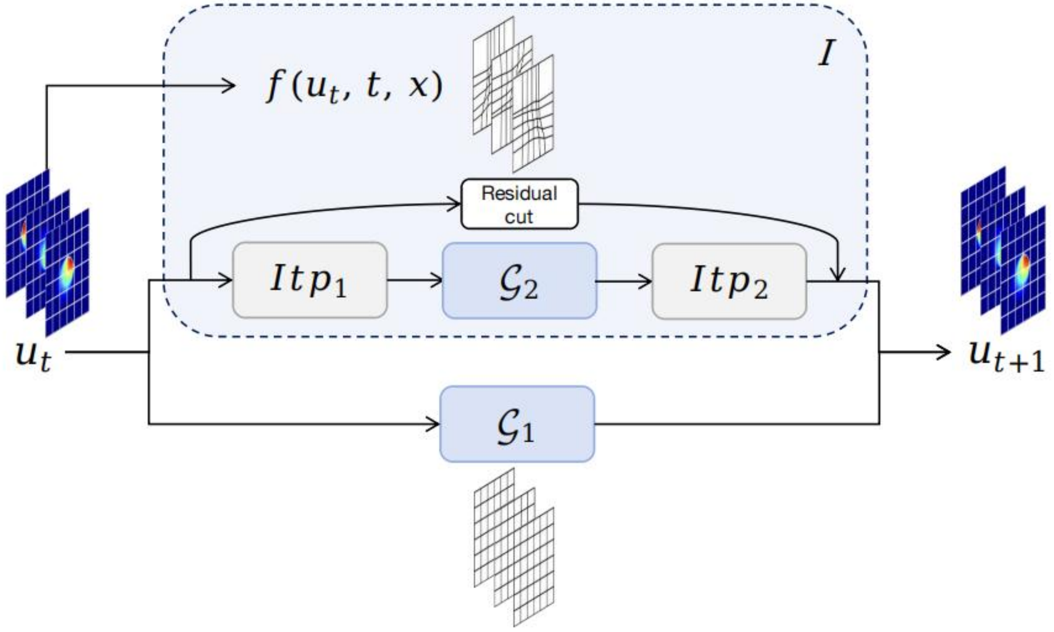
$$l = l_{equation} + \beta l_{bound} + \gamma l_{convex},$$
$$l_{equation} = \| \, |\Omega| \, \rho(\nabla_x \psi + x) det(H(\psi) + I) - \sigma \|,$$
$$l_{bound} = \|\nabla_x \psi(\partial\Omega)\|,$$
$$l_{convex} = \min\{0, 1 + \partial_{x_1}^2 \psi\}^2 + \min\{0, 1 + \partial_{x_2}^2 \psi\}^2.$$

- Sampling strategy: sampling probability proportional to M.



Moving mesh based neural PDE solver (MM-PDE)

- **Effecient message passing:**
- **Between time steps**: leverage the DMM for efficient node allocation.
- **From the original to the moving mesh**:
- (1) A learnable interpolation framework I: uses a residual cut network to convey information on original meshes and Itp_i to generate self-adapting interpolation weights. Pretrained on original data.
- (2) Two parallel branches that pass messages on the original and moving meshes respectively.
- $$\mathcal{M}(u_t) = \mathcal{G}_1(u_t) + I(\mathcal{G}_2, \tilde{f}, u_t)$$



Experiment

- Our method can generate and utilize moving meshes reasonably, and is superior in accuracy.

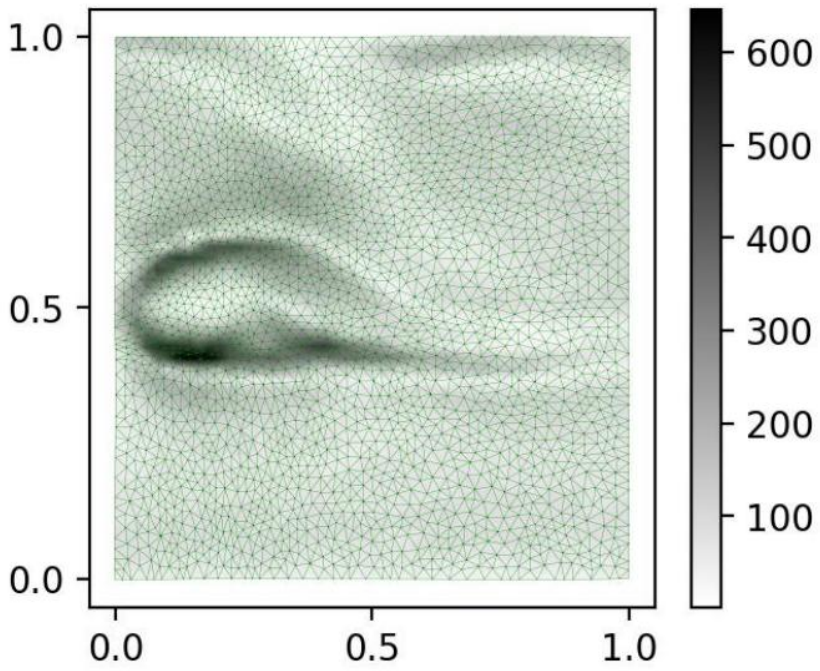


Table 1: Results of Burgers' equation and flow around a cylinder.

(a) Meshes from DMM on the Burgers' equation.

METRIC	ORIGINAL UNIFORM MESH	MESH FROM DMM
std	0.1027	**0.0469**
range	0.7524	**0.2061**

(b) MM-PDE and baselines on the Burgers' equation.

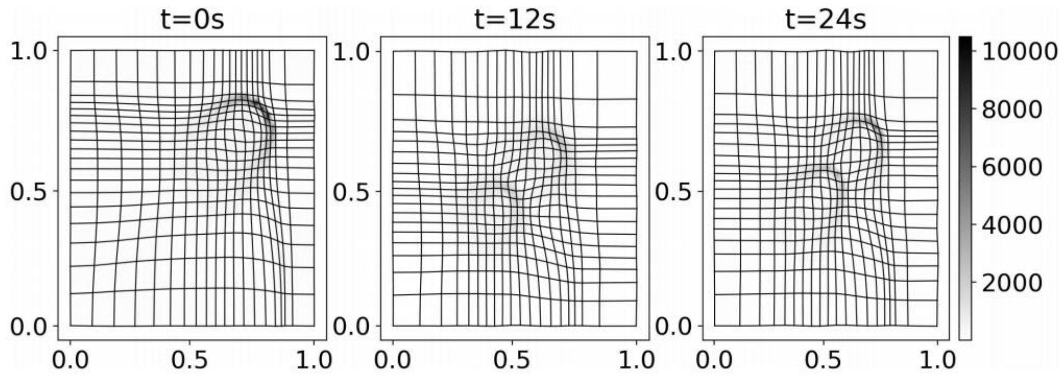
MODEL	ERROR	TIME(s)
MM-PDE	**1.04e-05**	0.5192
GNN	3.10e-05	0.3078
bigger GNN	1.19e-05	0.3298
CNN	1.26e-05	0.0027
FNO	1.18e-05	0.0159
LAMP	1.61e-05	1.4598

(c) Meshes from DMM on flow around a cylinder.

METRIC	ORIGINAL UNIFORM MESH	MESH FROM DMM
std	0.0136	**0.0094**
range	0.1157	**0.0733**

(d) MM-PDE and baselines on flow around a cylinder.

MODEL	ERROR
MM-PDE	**0.0846**
GNN	0.2892
bigger GNN	0.1548
CNN	-
FNO	-
LAMP	0.4040
Geo-FNO	0.2844



Experiment

Table 7: Test on data of different resolutions on the Burgers' equation.

Metric	Uniform 24×24 Mesh	24×24 Mesh from DMM
std	0.3315	0.1596
range	1.6879	0.7055

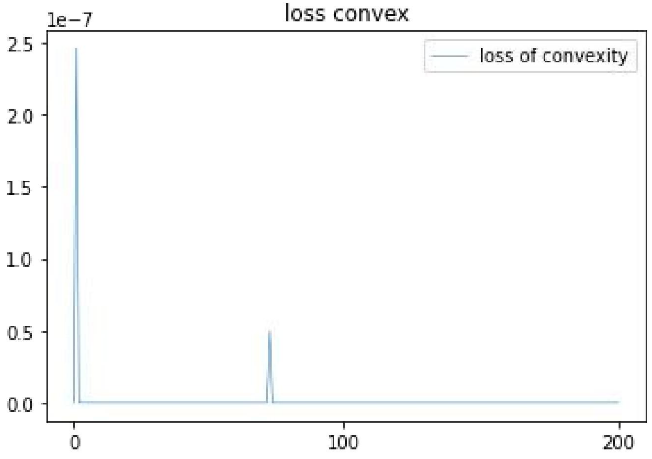
Metric	Uniform 96×96 Mesh	96×96 Mesh from DMM
std	0.0304	0.0136
range	0.3273	0.0858

Table 8: Test on data of different resolutions on the flow around a cylinder.

Metric	Original 2521 Mesh	3286 Mesh from DMM
std	0.0112	0.0077
range	0.1111	0.0665

Metric	Original 2521 Mesh	1938 Mesh from DMM
std	0.0163	0.0111
range	0.1488	0.0833

- DMM's ability to generate meshes of varying resolutions during inference.



- The mesh tangling is rare.

Table 9: Results of variants of MM-PDE.

MODEL	ERROR
MM-PDE	**6.63e-07**
no $\mathscr{G}_1$	1.258e-06
$\mathscr{G}_1 + \mathscr{G}_2$	8.852e-07
no Residual	8.467e-07
uniform mesh	1.324e-06

- A well-defined and suitable moving mesh is crucial. And our introduced ways of message passing are effective.

Thank you!

If you have any question, please feel free to contact us.
hupeiyan18@mails.ucas.ac.cn
yuwang5@microsoft.com