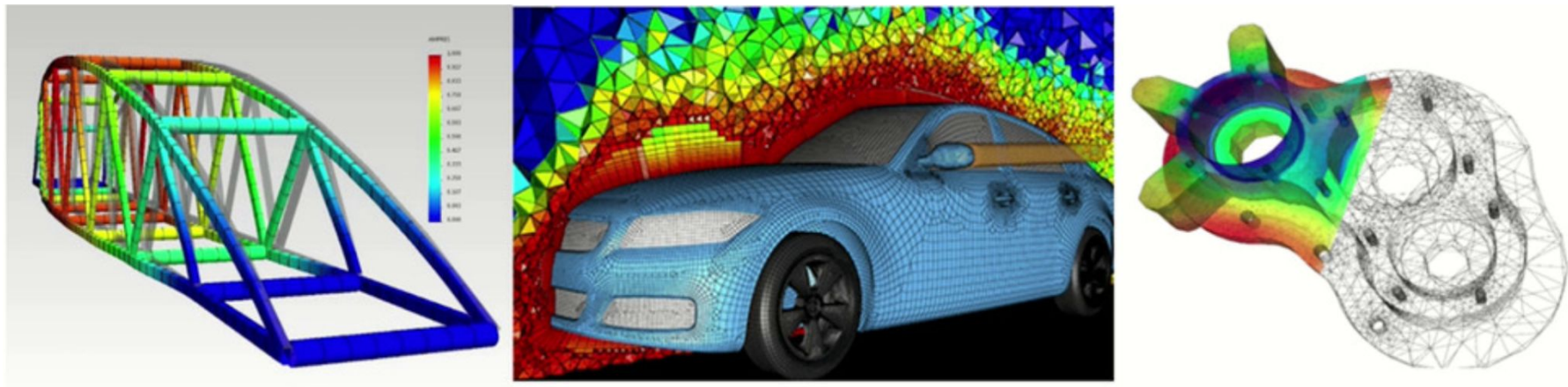


# **LEARNING MESH-BASED SIMULATION WITH GRAPH NETWORKS**

# Mesh-based representation

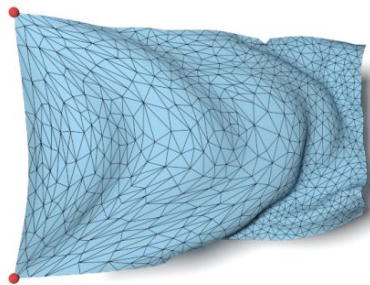
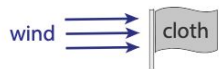


Представления на основе мешей часто используются в науке и инженерии

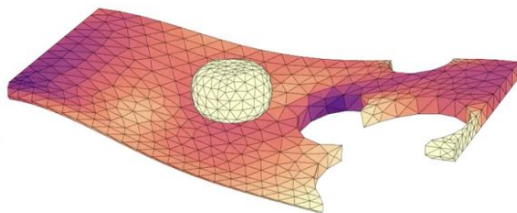
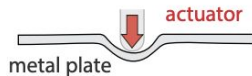
# Обучение mesh-based моделированию

- Хорошая обобщающая способность
- В 10 - 200 раз быстрее чем ground truth
- Подходит для абсолютно разных физических систем

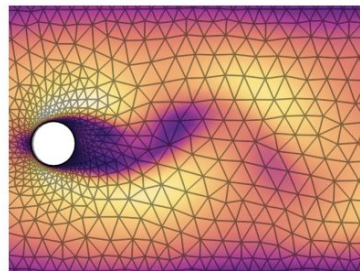
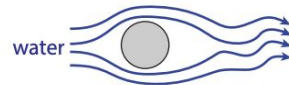
(a) FlagDynamic



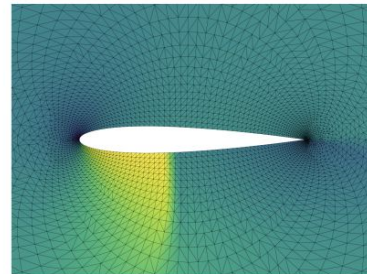
(b) DeformingPlate



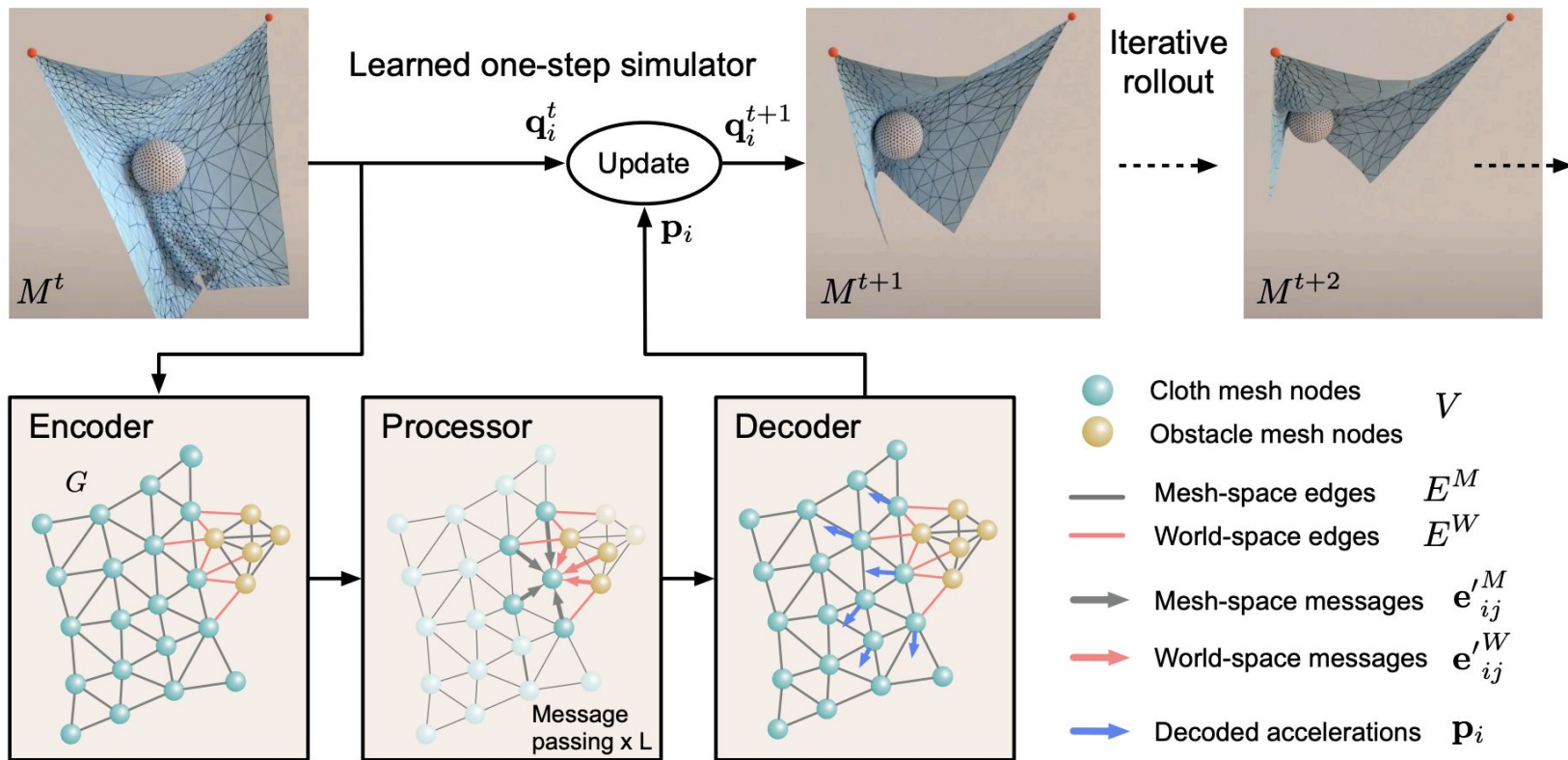
(c) CylinderFlow



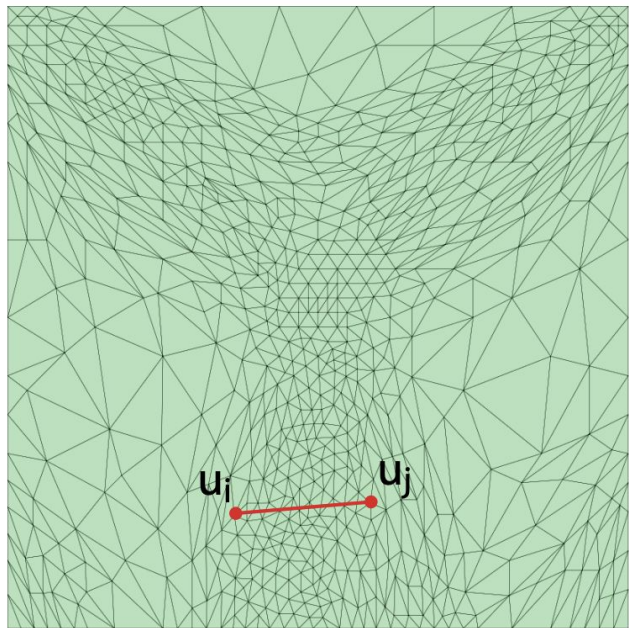
(d) Airfoil



# Архитектура модели

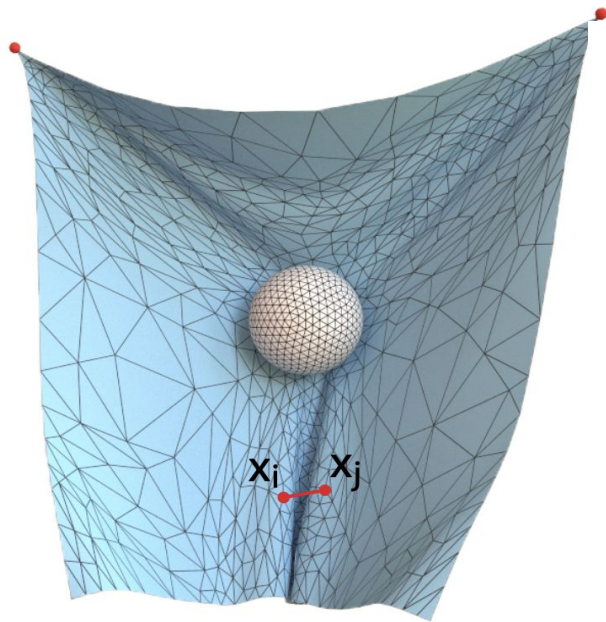


# Использование двойного пространства



mesh space  $u$

Необходим для понимания **внутренней** динамики

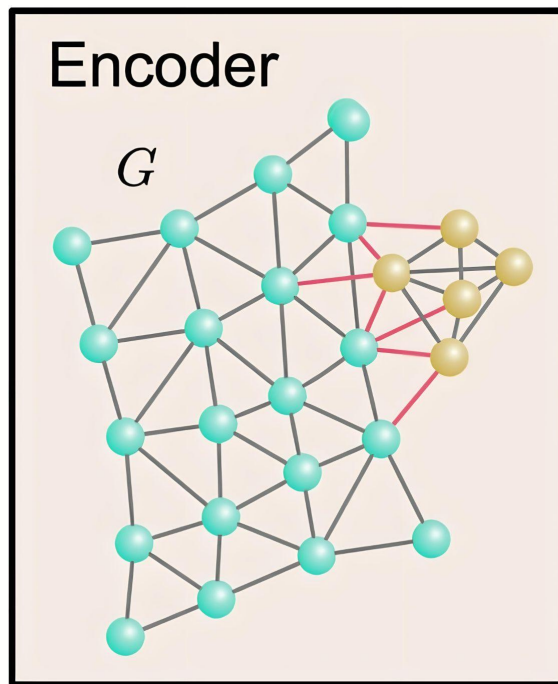
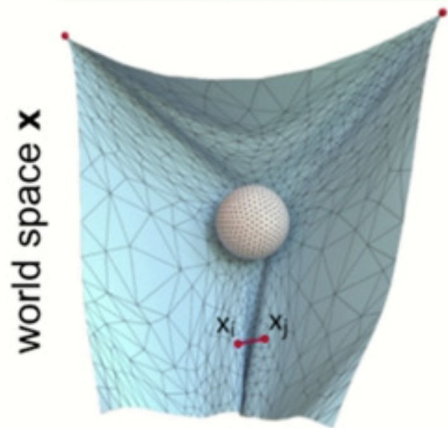
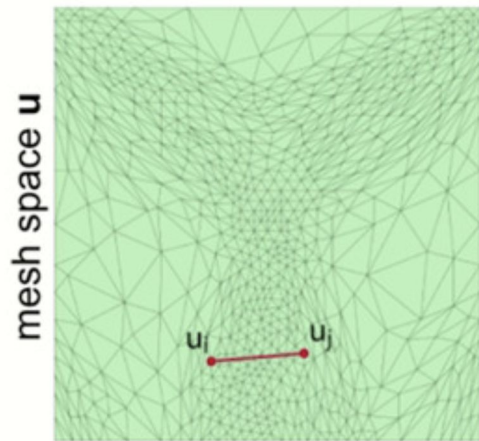






world space  $x$

Необходим для понимания **внешней** динамики: например, соприкосновений



# Архитектура Encoder



-  Cloth mesh nodes
-  Obstacle mesh nodes
-  Mesh-space edges
-  World-space edges

Mesh edge features:

$$\mathbf{x}_{ij} = \mathbf{x}_j - \mathbf{x}_i$$

$$\mathbf{u}_{ij} = \mathbf{u}_j - \mathbf{u}_i$$

World edge features

$$\mathbf{x}_{ij} = \mathbf{x}_j - \mathbf{x}_i$$

Node features

$$\mathbf{vel}_i$$

$$\mathbf{x}_i$$

$$\mathbf{u}_i$$

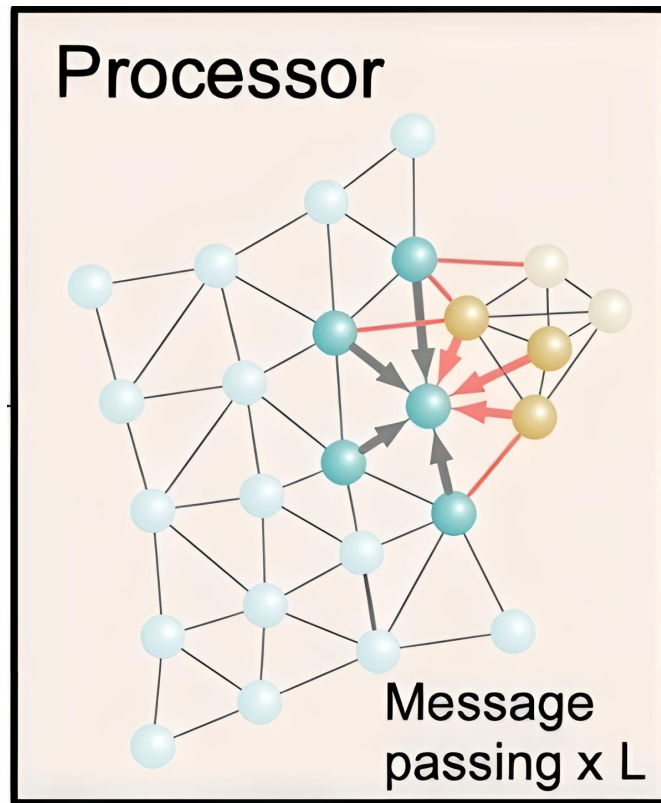
# Архитектура Processor

Состоит из  $L$  последовательных одинаковых блоков, каждый со своими весами

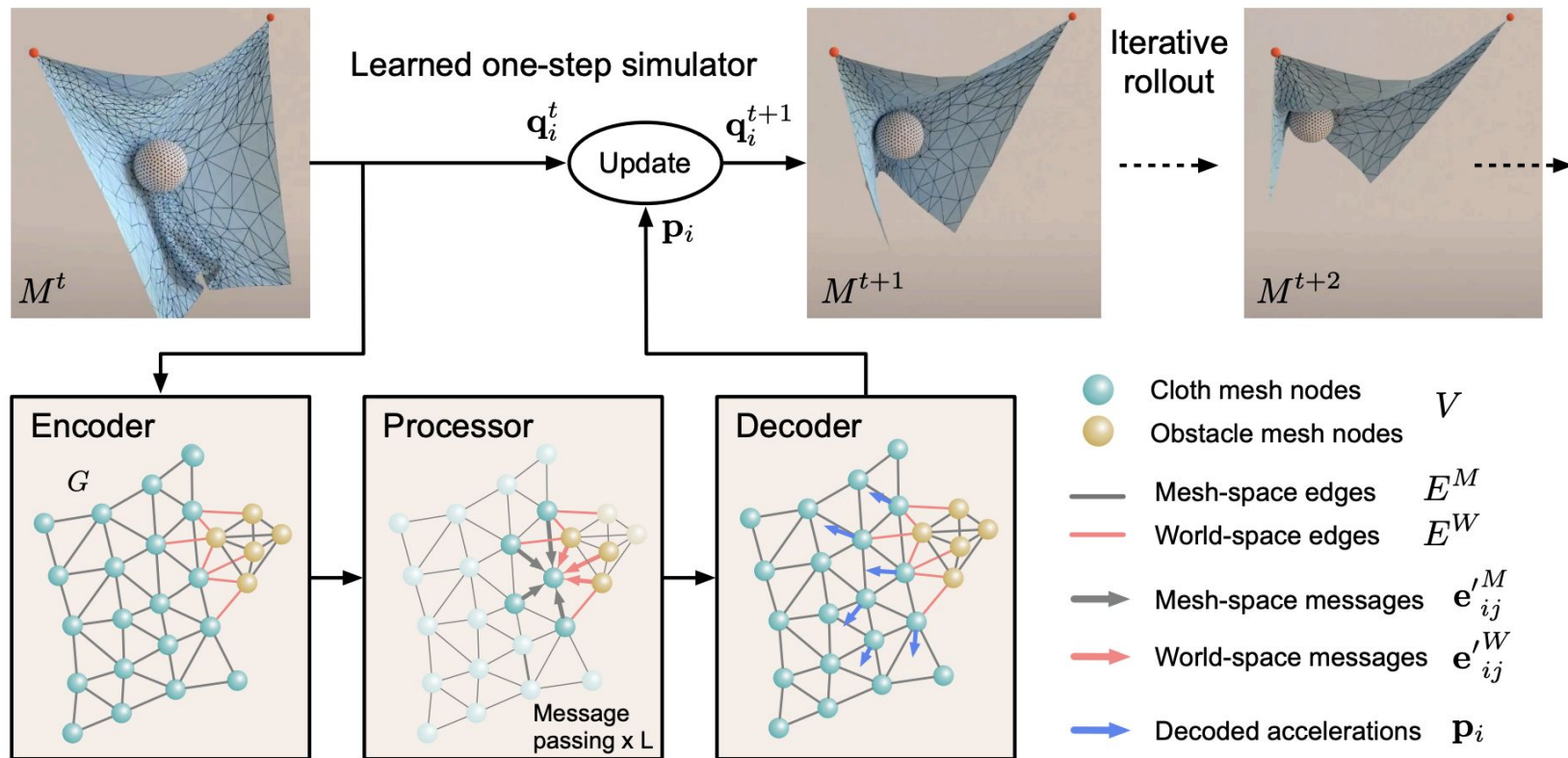
$$\mathbf{e}'_{ij}{}^M \leftarrow f^M(\mathbf{e}_{ij}^M, \mathbf{v}_i, \mathbf{v}_j) \quad \mathbf{e}'_{ij}{}^W \leftarrow f^W(\mathbf{e}_{ij}^W, \mathbf{v}_i, \mathbf{v}_j)$$

$$\mathbf{v}'_i \leftarrow f^V(\mathbf{v}_i, \sum_j \mathbf{e}'_{ij}{}^M, \sum_j \mathbf{e}'_{ij}{}^W)$$

$f^M, f^W, f^V$  - MLPs with a residual connection



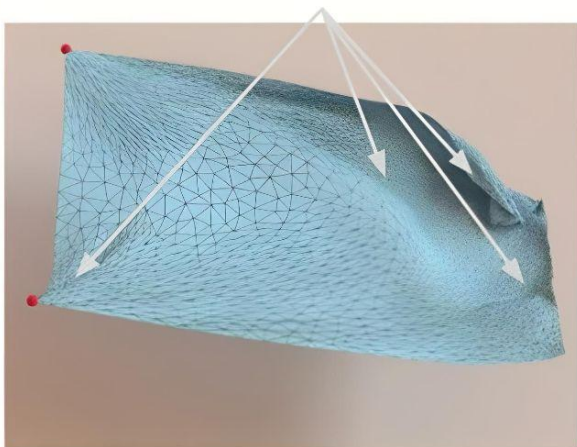
# Архитектура модели





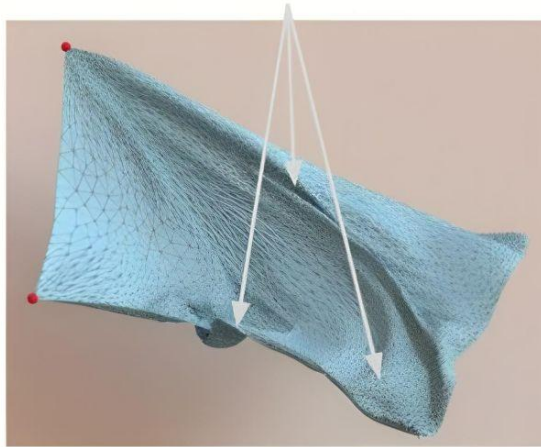
# Adaptive remeshing

Fine-scale regions at  $t_i$

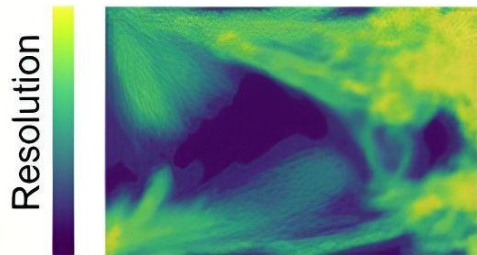


Time  
-->

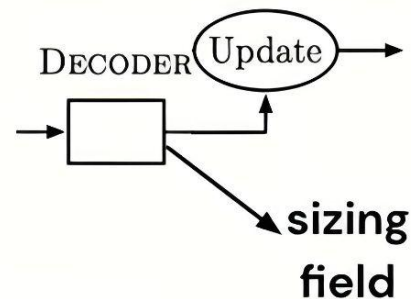
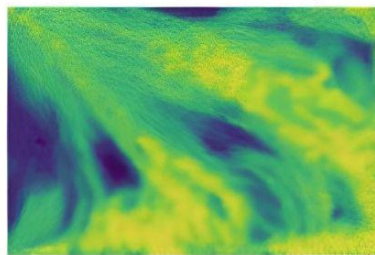
Fine-scale regions at  $t_j$



Sizing field at  $t_i$



Sizing field at  $t_j$



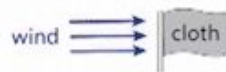
**Предсказываем Sizing  
Field и делаем remeshing**

# Результаты

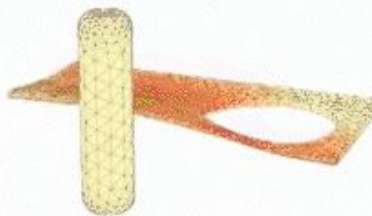
Particle-based  
fluids



Cloth simulation



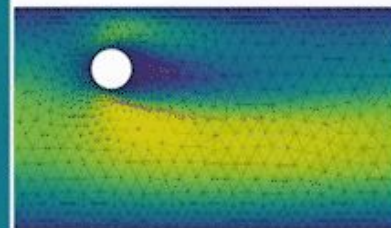
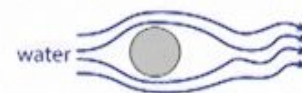
Structural mechanics



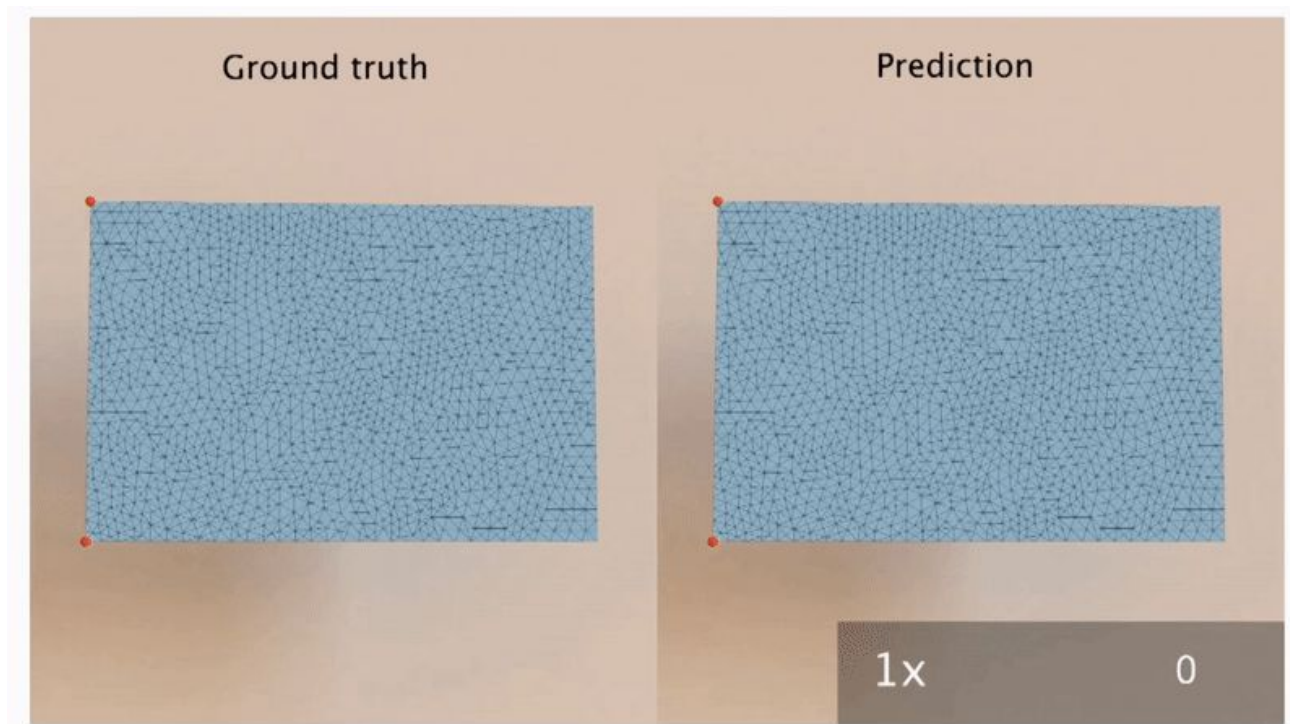
Compressible  
aerodynamics



Incompressible  
fluid simulation



# Стабильность при длительных симуляциях



# Высокая обобщающая способность



# Выводы

- Обученная модель обладает рядом преимуществ:
  - Адаптивность
  - Высокая скорость применения,
  - Большая обобщающая способность
- Перспективная для применения в научных и инженерных проблемах



**Практик-исследователь**

# LEARNING MESH-BASED SIMULATION WITH GRAPH NETWORKS

**Tobias Pfaff\*, Meire Fortunato\*, Alvaro Sanchez-Gonzalez\*, Peter W. Battaglia**

Deepmind, London, UK

{tpfaff, meirefortunato, alvarosg, peterbattaglia}@google.com

Написана в октябре 2020

Представлена на ICLR 2021, спотлайт

# Tobias Pfaff

 Research Scientist @ DeepMind

 ETH Zürich  UC Berkeley

 Simulation, Graphs, Fluids

Делал проекты в DoubleFine и Shinyshoe,  
работал Avametric



# Meire Fortunato



👛 Research Scientist @ DeepMind

🏫 Universidade Estadual de Campinas ➡  
UC Berkeley

💡 Deep Learning, Numerical Methods, Mesh generation

# Alvaro Sanchez Gonzalez

👛 Staff Research Engineer and Team Lead  
@ DeepMind

🏫 Universidad de Salamanca ➡ Imperial  
College London

💡 Simulation, Graphs, Fluids







# Peter Battaglia



 Senior Staff Research Scientist @  
DeepMind

 University of Rochester  University of  
Minnesota (психология)

 Graphs, Deep Learning

# Relational inductive biases, deep learning, and graph networks

**Peter W. Battaglia<sup>1\*</sup>, Jessica B. Hamrick<sup>1</sup>, Victor Bapst<sup>1</sup>,  
Alvaro Sanchez-Gonzalez<sup>1</sup>, Vinicius Zambaldi<sup>1</sup>, Mateusz Malinowski<sup>1</sup>,  
Andrea Tacchetti<sup>1</sup>, David Raposo<sup>1</sup>, Adam Santoro<sup>1</sup>, Ryan Faulkner<sup>1</sup>,  
Caglar Gulcehre<sup>1</sup>, Francis Song<sup>1</sup>, Andrew Ballard<sup>1</sup>, Justin Gilmer<sup>2</sup>,  
George Dahl<sup>2</sup>, Ashish Vaswani<sup>2</sup>, Kelsey Allen<sup>3</sup>, Charles Nash<sup>4</sup>,  
Victoria Langston<sup>1</sup>, Chris Dyer<sup>1</sup>, Nicolas Heess<sup>1</sup>,  
Daan Wierstra<sup>1</sup>, Pushmeet Kohli<sup>1</sup>, Matt Botvinick<sup>1</sup>,  
Oriol Vinyals<sup>1</sup>, Yujia Li<sup>1</sup>, Razvan Pascanu<sup>1</sup>**

<sup>1</sup>DeepMind; <sup>2</sup>Google Brain; <sup>3</sup>MIT; <sup>4</sup>University of Edinburgh

Статья предлагает базовую модель (Encoder - Processor - Decoder), которая используется в данной работе

---

## Combining Differentiable PDE Solvers and Graph Neural Networks for Fluid Flow Prediction

---

Filipe de Avila Belbute-Peres<sup>1†</sup> Thomas D. Economou<sup>2†</sup> J. Zico Kolter<sup>1,3</sup>

Статья с похожей идеей. Используется в качестве бейзлайна, работа развивает её идеи

---

## Learning to Simulate Complex Physics with Graph Networks

---

Alvaro Sanchez-Gonzalez<sup>\*1</sup> Jonathan Godwin<sup>\*1</sup> Tobias Pfaff<sup>\*1</sup> Rex Ying<sup>\*12</sup> Jure Leskovec<sup>2</sup>  
Peter W. Battaglia<sup>1</sup>

Статья от тех же авторов с похожими идеями, но в отличие от данной работы основной фокус на жидкостях

Two Minute Papers: <https://youtu.be/2Bw5f4vYL98>

# Развитие идей

106 цитирований

Полноценного продолжения нет

Конкурентов тоже не нашёл

Хочется увидеть в приложениях для 3D моделирования, позволит ускорить симуляции без видимых потерь качества

Дальнейшее исследование может включать более качественное изучение влияния отдельных идей на качество, есть подозрение, что не все и не всегда работают. Сравнение скорости тоже не очень честное, его можно улучшить



вместо хакера

Классное видео от Two Minute Papers:

<https://youtu.be/g7bEUB8aLvM>