Compact Neural Network Solutions to Laplace's Equation in a Nanofluidic Device

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

martinmagill.netlify.com

Martin Magill, Faisal Z. Qureshi, & Hendrick W. de Haan

Faculty of Science, University of Ontario Institute of Technology, Oshawa

cNAB LAB

faculty.uoit.ca/dehaan/cNAB.LAB/

Overview

• **Coarse-grained particle simulations** are important for the R&D of **nanofluidic devices**.

• Incorporating **electric fields** into these simulations can be **computationally expensive**.

• We illustrate the feasibility of using **deep neural networks** to solve and represent these fields.

• **Compact representation methods** could enable the use of **more sophisticated models**.

• We present this as a **new industrial application** of potential interest to the CDNNRIA community.

Background

• Nanofluidic devices can be used to detect, manipulate, or characterize biomolecules, like DNA.

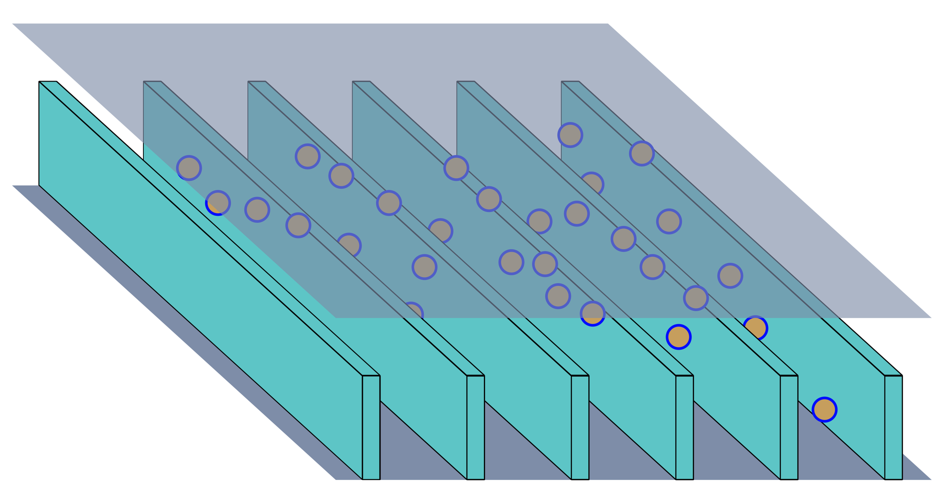
• Coarse-grained particle simulations can provide information that is inaccessible in experiments.

• These simulations can be parallelized perfectly, so efficiency is limited by available GPU memory.

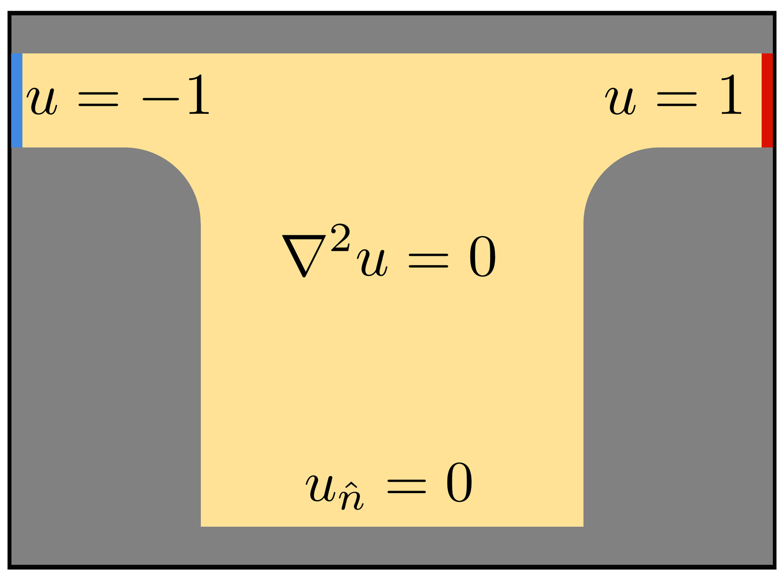
• Incorporating external force fields is memory-intensive, especially for more complicated forces.

• The deep neural network (DNN) method of solving partial differential equations (PDEs) has been shown to use less memory than traditional methods.

The slit-well device

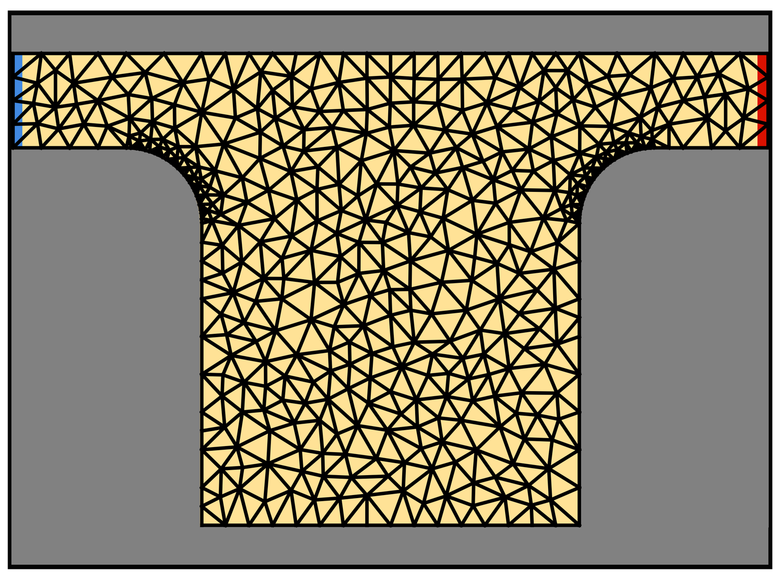


• The slit-well device consists of a series of deeper wells connected by shallower nanoslits. It can be used to separate nanoparticles or polymers by size.

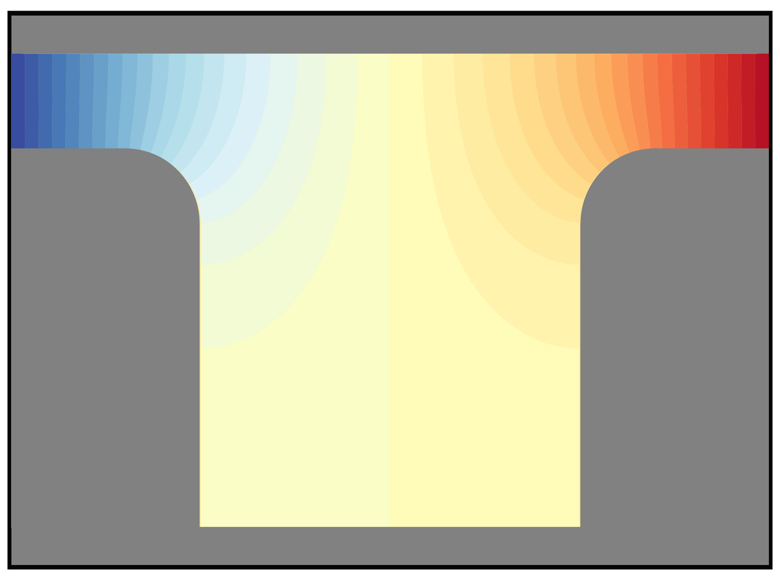


• The electric potential can be modelled by the Laplace partial differential equation (PDE).

The finite element method

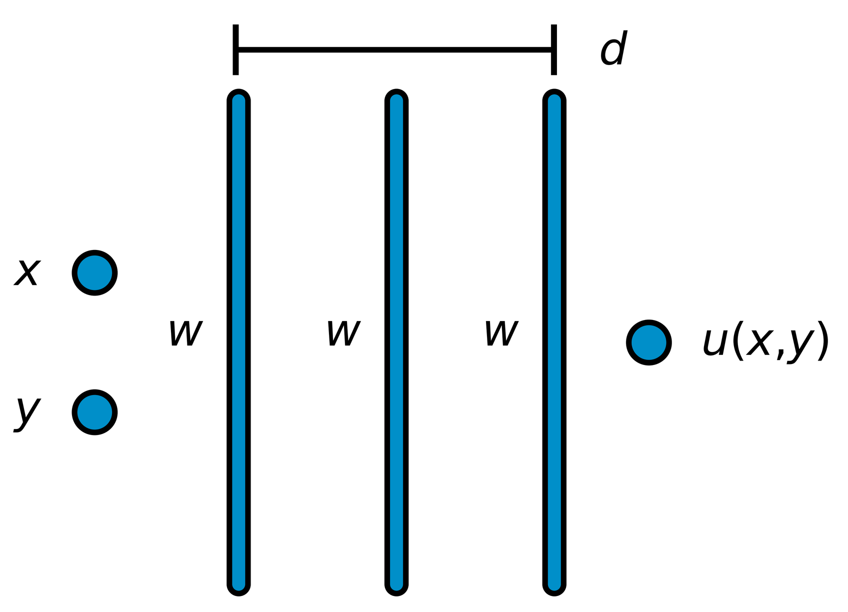


• In the finite element method (FEM), the domain is decomposed into a mesh of points. The round corners were approximated by linear segments.



• FEM is guaranteed to converge to the correct solution at high mesh resolutions. It was used to obtain an approximate ground truth solution.

The deep neural network method



• In the deep neural network (DNN) method, the PDE solution is approximated directly by a DNN.

• We used fully-connected tanh networks with d hidden layers of width w.

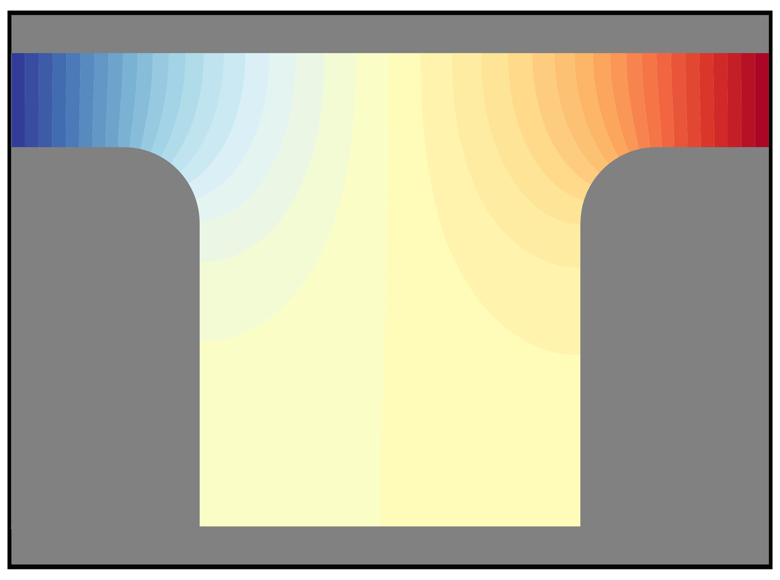
• The DNN was trained by stochastic gradient descent to minimize

$$\mathcal{L}[u] = \int_\Omega \|\nabla^2 u\|^2 dA + 10 \int_{\partial\Omega} \|B[u]\|^2 dS,$$

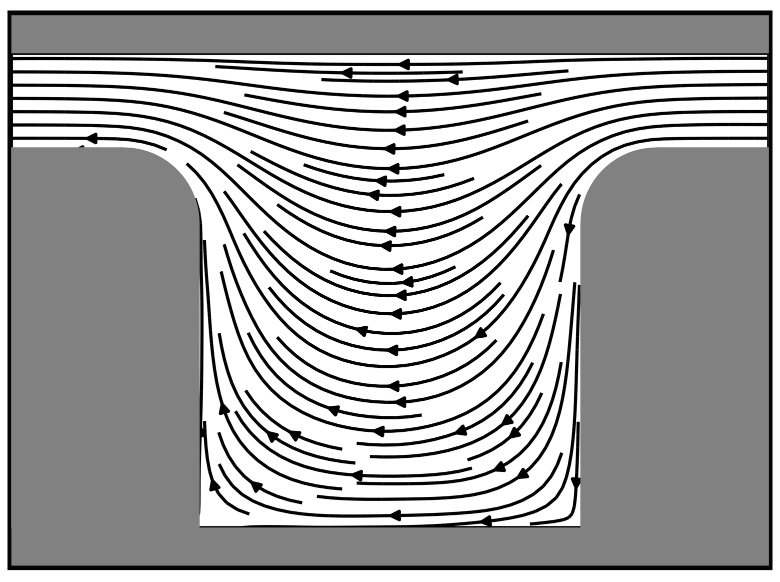
where $B[u]$ describes the boundary conditions.

• Training points were randomly sampled from the PDE domain to approximate these integrals.

• Rather than sampling the boundary separately, we approximated points near a given part of the boundary as lying on that segment.



• The solution above was obtained using a DNN with 3 layers of width 10.



• The electric field can be recovered directly from the DNN solution using automatic differentiation.

Solution accuracy and memory cost

• The FEM solutions used three double precision numbers and three integers per mesh point, so the memory cost for a mesh of N points is

$$2(3N) + 0.5(3N) = 7.5N,$$

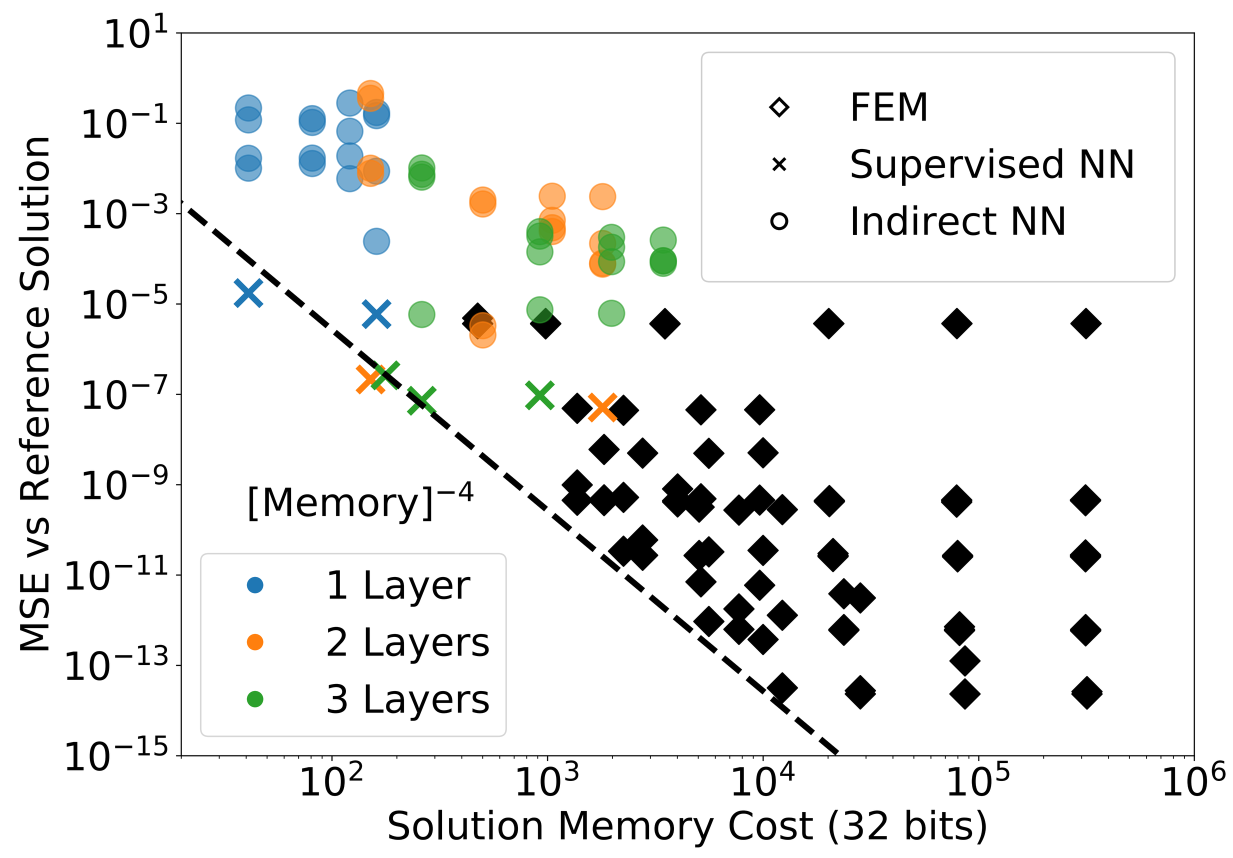
in increments of 32 bits, approximately.

• The DNNs used one single precision number for each weight and each bias, so the memory cost of a DNN of depth d and width w is:

$$3w + w(w+1)(d-1) + (w+1),$$

in increments of 32 bits.

• Below, we compare the memory consumption of the FEM and DNN methods against their mean squared error (MSE) relative to the approximate ground truth solution.



• The diamonds show FEM solutions with different mesh resolutions and different numbers of linear segments on the round corners.

• The dashed line indicates the balance of accuracy to memory cost among the best FEM solutions.

• The circles show DNN solutions with various widths and depths.

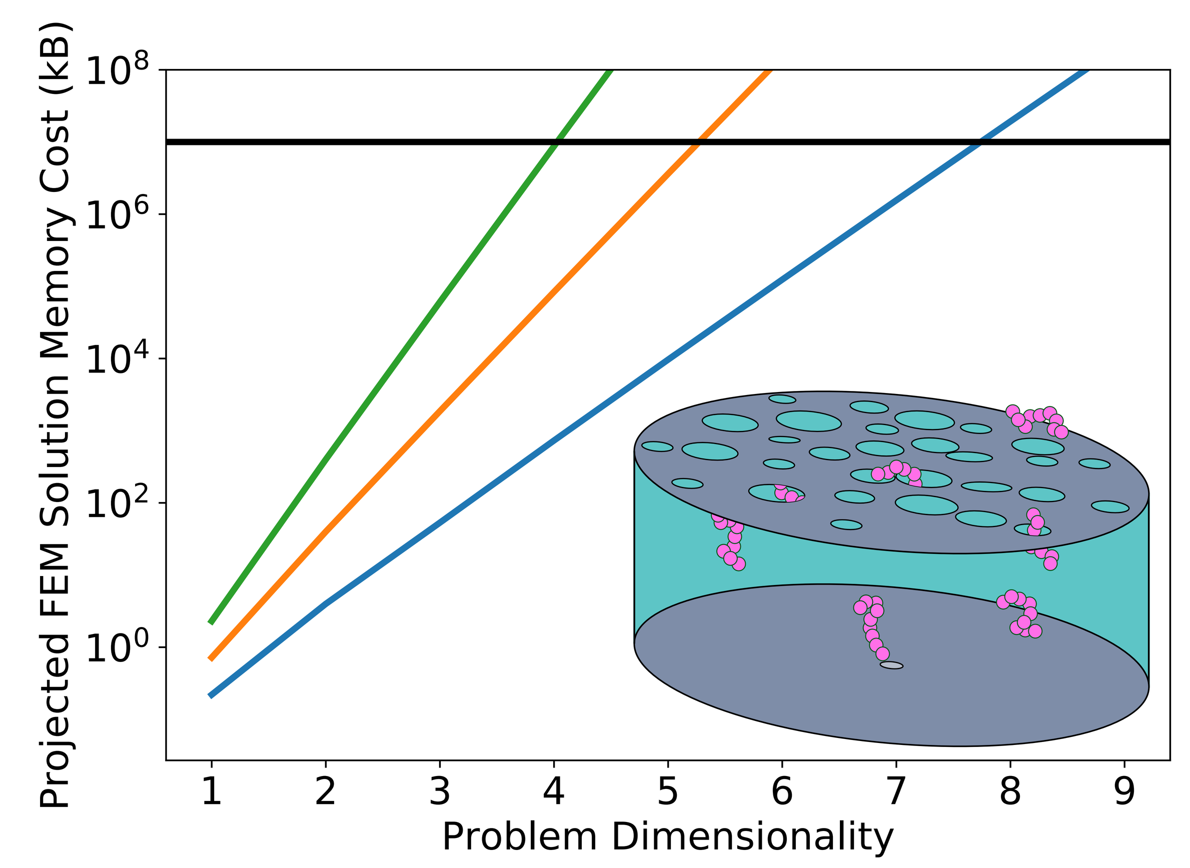
• Although the DNN method did not match the accuracy of FEM, it did attain accuracies sufficient for use in coarse-grained simulations.

• The crosses show **mimic** DNNs, which were trained directly to minimize MSE relative to the approximate ground truth solution.

• These mimics did achieve the same balance of accuracy and memory consumption as the FEM solutions.

• The improved performance of the mimics suggests the DNN method did not fully exploit the representational capacity of the networks, and that improved training algorithms might enhance performance.

Higher-dimensional problems



• Future work will use the DNN method for three-dimensional devices, time-varying force fields, and field-particle interaction models.

• The memory cost of mesh-based methods like FEM grows exponentially with the dimensionality of the PDE domain, whereas that of the DNN method grows at most polynomially [Grohs et al. 2018].

• Compact DNN representation techniques could reduce this cost even further, directly increasing overall simulation efficiency.

NSERC CRSNG · FEniCS PROJECT · NVIDIA CUDA · TensorFlow