Solving a Free Boundary System by Using Neural Networks

XINYUE (EVELYN) ZHAO[1], WENRUI HAO[2], BEI HU[1]

[1] Department of Applied and Computational Mathematics and Statistics, University of Notre Dame [2]Department of Mathematics, The Pennsylvania State University

INTRODUCTION

Free boundary problems deal with systems of partial differential equations, where the domain boundaries are apriori unknown. Due to this special characteristic, it is challenging to solve the free boundary problems either theoretically or numerically. We develop a novel approach for solving a modified Hele-Shaw problem based on **boundary integral method** and **neural network discretization**. The existence of the numerical solution under this new scheme is established theoretically. We also numerically verify this approach by computing the symmetry-breaking solutions which are guided by the bifurcation analysis near the radially-symmetric branch. Moreover, we further verify the capability of this approach by computing some non-radially symmetric solutions which are not characterized by any theorems.

THE MODEL PROBLEM

We need to solve both the unknown function σ and the unknown domain Ω of the following system:

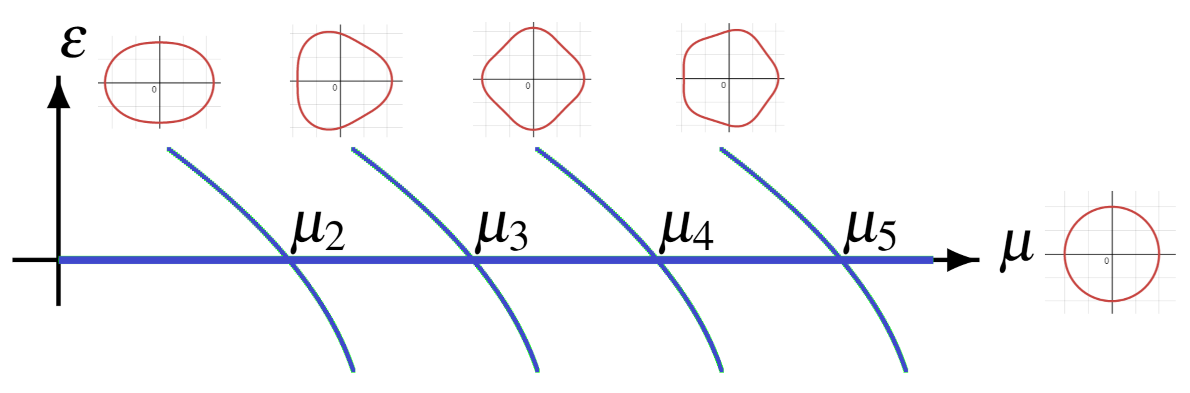
$$\begin{cases} -\Delta\sigma = -\sigma & \text{in } \Omega, \\ \sigma = \mu + \kappa & \text{on } \partial\Omega, \\ \frac{\partial \sigma}{\partial n} = \beta & \text{on } \partial\Omega, \end{cases} \qquad (1)$$

where μ and β are constants, and κ denotes the mean curvature of the domain boundary $\partial\Omega$.

THEORETICAL RESULTS

- System (1) admits a unique radially symmetric solution $(\sigma_S(r), R_S)$.
- For each $n \geq 2$, there exists a μ_n such that at each $\mu = \mu_n$, a symmetry-breaking solution branch bifurcates from the radially symmetric solution.
- Furthermore, the free boundary of the symmetry-breaking bifurcation solution is

$$r = R_S + \varepsilon\cos(n\theta) \text{ with } |\varepsilon| \ll 1.$$



BOUNDARY INTEGRAL METHOD

- Denote $G_1(\boldsymbol{x},\boldsymbol{y})$ by the fundamental solution of $(-\Delta+1)\sigma = 0$. In 2D, $G_1(\boldsymbol{x},\boldsymbol{y}) = \frac{i}{4}H_0^{(1)}(i|\boldsymbol{x}-\boldsymbol{y}|)$.
- Since $G_1'(r) = O(r^{-1})$ is strongly singular, we further introduce $Q(r) = \frac{1}{r}\left(G_1'(r) + \frac{1}{2\pi r}\right)$.
- Using Green's Theorem, and substituting into the boundary conditions from (1), we derive, for each $\boldsymbol{x} \in \partial\Omega$,

$$\int_{\partial\Omega}\left[\beta G_1(\boldsymbol{x},\boldsymbol{y}) - \left((\mu+\kappa(\boldsymbol{y}))Q(|\boldsymbol{x}-\boldsymbol{y}|) - \frac{\kappa(\boldsymbol{y})-\kappa(\boldsymbol{x})}{2\pi|\boldsymbol{x}-\boldsymbol{y}|^2}\right)(\boldsymbol{y}-\boldsymbol{x})\cdot\boldsymbol{n}_y\right]\mathrm{d}S_y = 0. \qquad (2)$$

- **At the expense of a singular kernel, the advantage of applying BIM is that the dimensionality of the problem is reduced by one**

- In 1D, we write everything in polar coordinates, and denote the boundary points $\boldsymbol{x}$ and $\boldsymbol{y}$ in (2) as:

$$\boldsymbol{x} = (R(\hat\theta)\cos(\hat\theta), R(\hat\theta)\sin(\hat\theta));$$
$$\boldsymbol{y} = (R(\theta)\cos(\theta), R(\theta)\sin(\theta)).$$

- Assume the unknown free boundary is $\partial\Omega : r = R(\theta)$, and denote $D[R] = |\boldsymbol{x}-\boldsymbol{y}|$.
- Based on (2), if we define an operator $\mathscr{L}$ by

$$\mathscr{L}[R](\hat\theta) \triangleq \int_{\partial\Omega}\left[\beta G_1(D[R]) - \left((\mu+\kappa(\boldsymbol{y}))Q(D[R]) - \frac{\kappa(\boldsymbol{y})-\kappa(\boldsymbol{x})}{2\pi(D[R])^2}\right)(\boldsymbol{y}-\boldsymbol{x})\cdot\boldsymbol{n}_y\right]\mathrm{d}S_y,$$

then we have $\mathscr{L}_\tau[R](\hat\theta) \equiv 0$ for each $\hat\theta \in [0,2\pi]$.

- Since the kernel is singular, we introduce $D_\tau[R] = \sqrt{(D[R])^2+\tau^2}$, and define $\mathscr{L}_\tau$ by

$$\mathscr{L}_\tau[R](\hat\theta) \triangleq \int_{\partial\Omega}\left[\beta G_1(D_\tau[R]) - \left((\mu+\kappa(\boldsymbol{y}))Q(D_\tau[R]) - \frac{\kappa(\boldsymbol{y})-\kappa(\boldsymbol{x})}{2\pi(D_\tau[R])^2}\right)(\boldsymbol{y}-\boldsymbol{x})\cdot\boldsymbol{n}_y\right]\mathrm{d}S_y.$$

- $\mathscr{L}_\tau[R] \to \mathscr{L}[R]$ when $\tau \to 0$, and **we have a non-singular kernel** when $\tau > 0$.
- If $r = R(\theta)$ is the unknown free boundary, then $\mathscr{L}_\tau[R](\hat\theta) \approx 0$ for each $\hat\theta \in [0,2\pi]$.

NEURAL NETWORKS

- Approximate $R(\theta)$ by a single hidden layer neural network (denote the parameter set by $\mathscr{X}$):

$$R(\theta) \approx \sum_{i=1}^{N} a_i\Psi(b_i\theta + c_i) + d \triangleq \rho(\theta;\mathscr{X}).$$

- Loss function $F(\mathscr{X},\hat{\boldsymbol{\theta}}) \triangleq \frac{1}{m}\sum_{i=1}^{m}\left(\mathscr{L}_\tau[\rho](\hat\theta_i)\right)^2 + \sum_{\alpha=0}^{2}\left(D^\alpha\left(\rho(0;\mathscr{X}) - \rho(2\pi;\mathscr{X})\right)\right)^2$.
- $\mathscr{X}$ is obtained via $\min_{\mathscr{X}} J(\mathscr{X}) \triangleq \mathbb{E}_{\hat{\boldsymbol{\theta}}}\left[F(\mathscr{X},\hat{\boldsymbol{\theta}})\right]$.

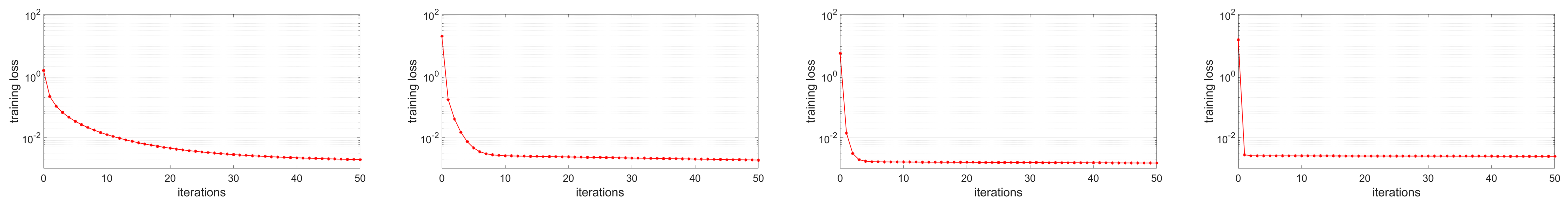
ALGORITHM

A SGD method to compute $\mathscr{X}$:

1: Choose an initial guess $\mathscr{X}_1$
2: **for** $k = 1,2,\cdots$ **do**
3: Generate m random points $\hat{\boldsymbol{\theta}}_k = (\hat\theta_{k,i})_{i=1}^m$;
4: Calculate loss function at randomly sampled points $F(\mathscr{X}_k,\hat{\boldsymbol{\theta}}_k)$;
5: Compute $G(\mathscr{X}_k,\hat{\boldsymbol{\theta}}_k) = \nabla_{\mathscr{X}}F(\mathscr{X}_k,\hat{\boldsymbol{\theta}}_k)$;
6: Set $\mathscr{X}_{k+1} = \mathscr{X}_k - \alpha_n G(\mathscr{X}_k,\hat{\boldsymbol{\theta}}_k)$;
7: **end for**

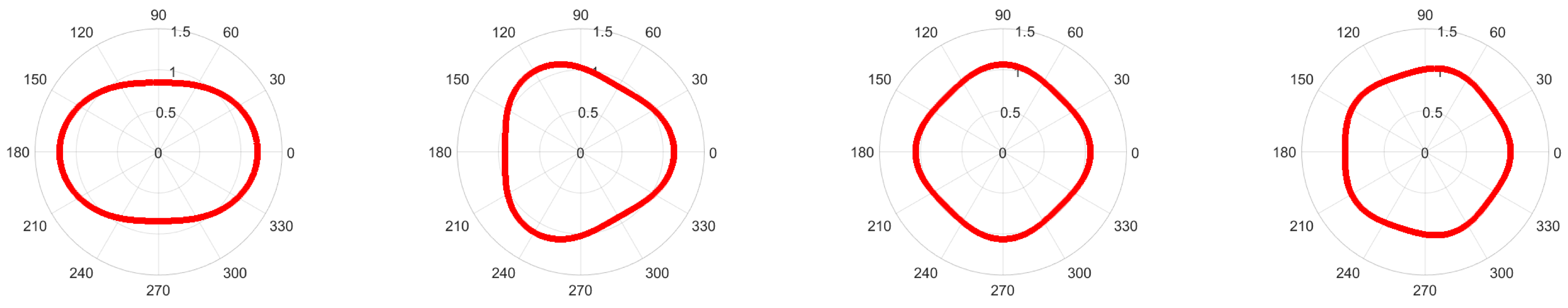
SIMULATION RESULTS

1. Verification of the scheme near bifurcation points



(a) $n=2$ bifurcation, $\mu = 14.6$. **(b)** $n=3$ bifurcation, $\mu = 28.6$. **(c)** $n=4$ bifurcation, $\mu = 47.0$. **(d)** $n=5$ bifurcation, $\mu = 70.0$.

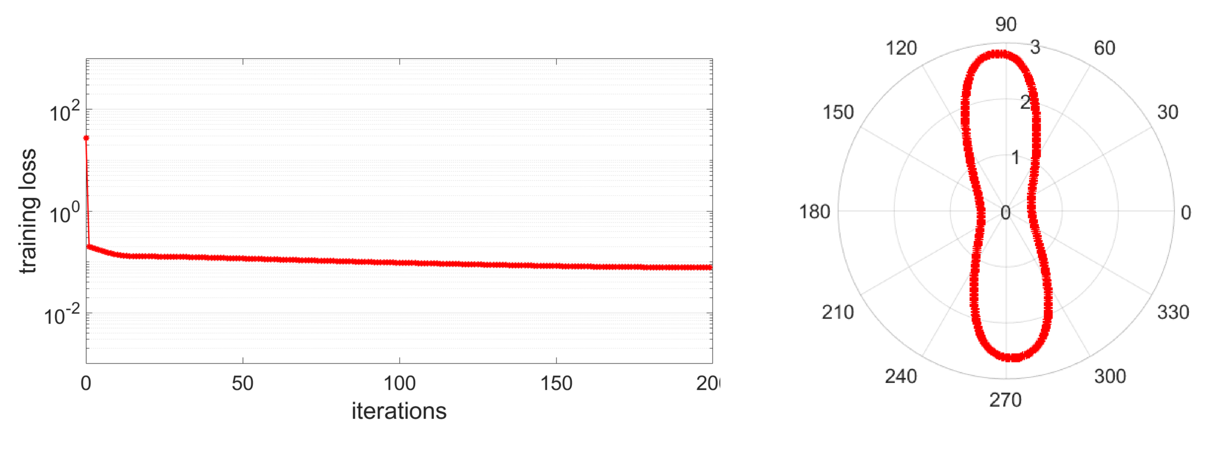
Figure 1: Training loss.



(a) $n=2$ bifurcation, $\mu = 14.6$. **(b)** $n=3$ bifurcation, $\mu = 28.6$. **(c)** $n=4$ bifurcation, $\mu = 47.0$. **(d)** $n=5$ bifurcation, $\mu = 70.0$.

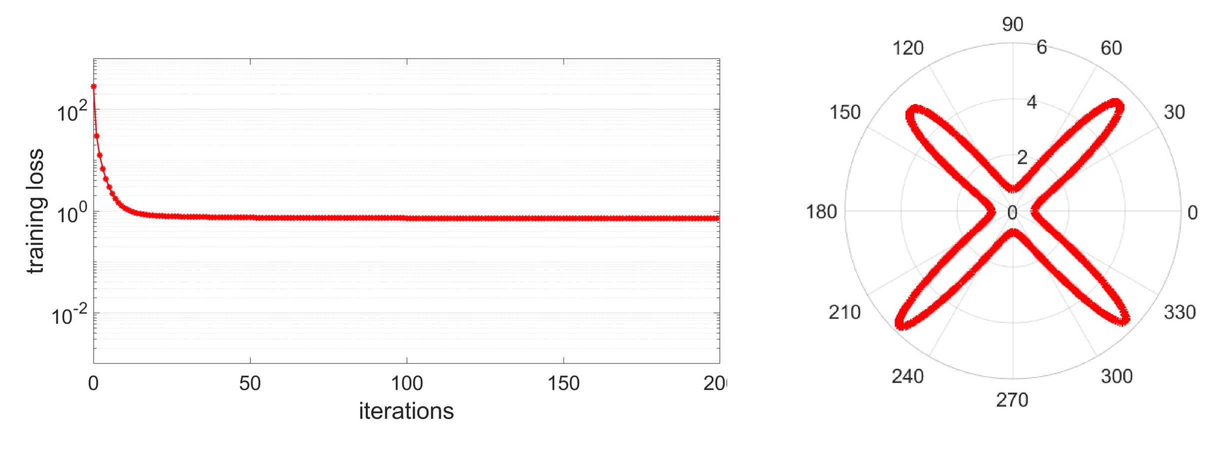
Figure 2: Contour plot of nonradially symmetric solutions in different bifurcation branches.

2. Other non-radially symmetric solutions



(a) Training loss. **(b)** Contour plot.

Figure 3: Non-radially symmetric solution with 2 fingers.



(a) Training loss. **(b)** Contour plot.

Figure 4: Non-radially symmetric solution with 4 fingers.

REFERENCES

[1] Zhao, X. E., Hao, W., and Hu, B., *Convergence analysis of neural networks for solving a free boundary system,* arXiv preprint , arXiv:2011.00315.

[2] Hao, W., Hu, B., Li, S., and Song, L., *Convergence ofboundary integral method for a free boundary system,* Journal ofComputational and Applied Mathematics, Vol. 334 (2018), pp. 128-157.

[3] Hornik, K., *Approximation capabilities of multilayer feed forward networks,* Neural networks, Vol. 4 (1991), pp. 251-257.