AUTOMATED AERODYNAMIC DESIGN USING NEURAL NETWORKS AND GRADIENT DECENT

Oliver Hennigh
Mexico
Guanajuato, Gto, Mexico
loliverhennigh101@gmail.com

ABSTRACT

Optimizing an objects geometry to have desired fluid flow properties has applications in many engineering setting such as aeornotical, automotive, and chemical engineering. In this paper, we propose a novel method that makes use of deep neural network and gradient decent to perform aerodynamic optimization of 2D and 3D airfoils in steady state flow. Our approach works by training a neural network to approximate the fluid simulation and compute values such as drag and lift. Then we use gradient decent on the parameter space of the airfoils to maximize the lift drag ratio at diffrent angles of attack. Because the network can be evaluated orders of magnatude faster then the flow solver and the gradient decent allows optimization to be performed in few iterations, we are able to find optimized designs with desired properties in merr minutes vers the several days required by other flow solver based search methods. We also enphasize that the methodology presented here can be used on many other automated design problems and the intuition behind our belief that why we believe that the Present and inituition of this and our belief that this is a fundamentaly better way to perform.

1 INTRODUCTION

(paragraph about why optimization is important)

Automated Design is the process by which a object is designed by a computer to meet or maximize some objective. This is typicaly performed by modeling the system and then exploring the space of designs to find a solution with good performance be it a veighcal with low drag or a . The classic example of this is the 2006 NASA ST5 spacecraft antenna designed by a evolutionary algorithm to create the best radiation pattern. More recently there has been (Flow Sculpter) and (optical computer). While there have been some significant sucesses in this feild the dream of true automated is still far from realized. The main challanges faced are heavy computational requirements for accuratly modeling the physical system and often exponentialy large search space. These two problems negatively complement eachother making the computation requirements intractable for even simple problems. For example, in the realively simple flow problems explored in this work an optimized flow solver running on modern gpus requires around 5 minutes to perform each simulation. Given that around 1,000 designs need to be tested to acheive reasonable performance this results a total computation time of 3 days on a single GPU. Increassing the resolution of the simulation or expanding the parameter space quickly make this an unrealizable problem without considerable resources.

Our approach works to solve the current problems of automated design in two ways. First, we learn a computationaly effeicent representation of the physics on a neural network. This trained network can be used to estimate the steady state flow for several orders of magnatude less time. Second, we use the differentiable nature of the trained network to get a gradient on the parameter space when performing optimization. This allows significanly better exploration of the parameter space and offers a scalable solution masive parameter spaces. These two abilitys of our method overcome the present difficulties with automated design and allow the previous mention 3 day optimization to be run in only 10 mins. While our method is centered around steady state fluid flow it is clear that the same approach is applicable to a wide variety of automated design problems.

In this work we look at optimizing an airfoil shape in a relatively low viscosity fluid and multiple angles of attack (angle of wing with respect to flow). We choose this setting to test our method because there has been numerous other works in this area.

This work has the following contributions.

- We present a novel way to use neural networks to optimize object geometry in steady state fluid flow. We only look at fluid flow if this paper however in principle this method could be applied to optimizing design for other problems where neural network suragate models are used.
- We offer a new network architeture for predicting steady state fluid flow that vastly out performs previous models.

2 RELATED WORK

This is touches on several diffrent subjects

2.1 NEURAL NETWORK BASED SURRAGATE MODELS

Our method make use of a network that takes in boundary conditions and subsequently predicts the steady state flow around these boundarys. This original idea was first presented in (Guo et al., 2016). Our flow prediction network has several key difference to this original work. First, we heavily improve the network architecture by keeping the network all convolutional and taking advantage of both residual connections and a U-Network architeture. This proved to drasticaly improve accuracy while maintain fast computation. Second, it takes in the binary representation of the boundary conditions instead of the Signed distance map. We found that with our improved network architecture, we over came the issure in their work using such a representation of the boundary.

2.2 AUTOMATED DESIGN OPTIMIZATION OF AIRFOILS

There has been substantial work to date in automated aerodynamical design for use in aeronotical and automotice applications. The standard

Automotive Aerodynamic Design Exploration Employing New Optimization Methodology Based on CFD

Adaptive Shape Parameterization for Aerodynamic Design

Pros and Cons of Airfoil Optimization 1

Multi-level CFD-based Airfoil Shape Optimization With Automated Low-fidelity Model Selection

A Surface Parameterization Method for Airfoil Optimization and High Lift 2D Geometries Utilizing the CST Methodology

A Universal Parametric Geometry Representation Method CST

3 GRADIENT DECENT ON PARAMETER SPACE

Our method We look at the two network architetures of inter and explain how they can be used in tandum perform boundary optimization.

3.1 FLOW PREDICTION NETWORK

As mentioned above, we have made significant changes to prior network architecture design . A figure illustrating our networn can be found here. This model resembols the u-network architeture seen here with a series skip after each down sample. The advantages of this style of network are its high performance on image to image type tasks, trainablility on small datasets, and fast evaluation time in comparison to networks with large fully connected layers. The trainability on small datasets make this particulary effective on predicting steady state flow. Creating datasets of flow simulation to train on can be one of the most difficult aspects of this work and requires considerable computation. Our network is able to train on realively small datasets of only 4,000 flow simulation in comparision to the 100,000 required in previous work predicting steady state flow.

4 EXPERIMENTS

In the following sections we subject our method and model to a variety of test. The goal the goal being first to determine how accuraty and fast our model can predict steady state flow. In particular, what its accuracy in predicting values such a drag and lift as these are important quantitize in our optimization. Second, we investigate how effective our gradient decent optimization is. This line of tests compares our method to other none gradient based search techniques and illustrates what is happening in the optimization processes looks like. For example, what does the gradient surface look like.

4.1 FLOW PREDICTION ACCURACY

As mentioned before, our model significantly out performs priour work predicting steady state fluid flow.

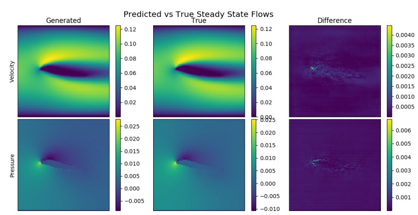

Figure 1: Flow

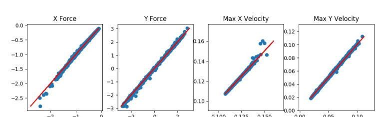

Figure 2: Flow

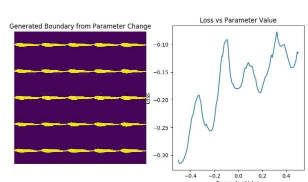

Figure 3: Flow

4.2 AUTOMATED DESIGN OF 2D AND 3D AIRFOILS

5 CONCLUSION

In this work we hav

Table 1: Sample table title

Batch Size	1	2	4	8	16
Flow Net 512^2	0.150 sec	0.101 sec	0.077 sec	0.065 sec	0.058 sec
Param Net 512^2	0.083 sec	0.045 sec	0.026 sec	0.015 sec	0.011 sec
Learn Step 512^2	0.494 sec	0.345 sec	0.270 sec	0.231 sec	Nan
Flow Net 144^3	0.826 sec	0.686 sec	0.627 sec	0.623 sec	Nan
Param Net 144^3	0.195 sec	0.144 sec	0.119 sec	0.106 sec	0.093 sec
Learn Step 144^3	3.781 sec	Nan	Nan	Nan	Nan

6 Sample Tables

Acknowledgments

This work was made possible through the `http://aigrant.org` run by Nat Friedman and Daniel Gross. This work would not have be possible without this very genourous support.

References

Xiaoxiao Guo, Wei Li, and Francesco Iorio. Convolutional neural networks for steady flow approximation. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 481–490. ACM, 2016.