# Weekly Thesis Presentation

## A project on Non-linear loudspeaker estimation

01.07.2024

Johannes Nørskov Toke

# Contents

# Contents (ii)

# 1 Theoretical considerations

# Model approaches

In data driven modelling 3 system classes of systems are often used to describe the how informed the model is being:

**Blackbox**

using no apriori information on model structure e.g. MLE based methods such as: Universal aproximator regression, sparse regression such as sindy and symbolic regression

**Graybox**

using some apriori information for some model structure e.g. Scientific ML such as **PINN's**,neural odes, **NN - injected models**

**Whitebox**

using only apriori information to infer the model structure e.g. physics-based models

# Universal approximators

A Universal Approximator ($\mathcal{U}$) is a function which is proven to estimate almost all functions to an arbitrary precisoin. Currently at least 8 types of $\mathcal{U}$ have been proven:

**Taylor - f must be k-times differentiable at a**

$f(x) = \sum_{i=0}^{k} c_i (x-a)^i + R_k(x) \text{where} c_i = \frac{1}{i!} \frac{\mathrm{d}^i}{\mathrm{d}x^i} f(x)|_{x=a}$ and $R_k(x) = o\left(|x-a|^k\right)$ which is the residual term.

**Fourier Series – f is piecewise C¹ and 2π-periodic**

$f(x) = A_0 + \sum_{n=1}^{N} \left[A_i \cos\left(\frac{2\pi i}{T} x + \phi_i\right)\right]$ where $T$ is the period of f,$A_i$ is the amplitude and $\phi_i$ is the phase of the i'th harmonic component.

# Universal approximators (ii)

**Stone,Weierstrass - f must be continuous real-valued function defined on an interval [a,b]**

**kolmogorov,Arnold - f must be continuous multivariate function**

**Funahashi,Hornick and Cybenko - single layer MLP**

$f_{\text{NN}}(x) = \sigma(\boldsymbol{A_1 X} + \boldsymbol{b_1})$ where $\boldsymbol{A}_1$ is $\mathbb{R}^m$

**Park et al. - MLP of width max(n+1,m)) for $\mathbb{R}^n \to \mathbb{R}^m$**

$f_{\text{NN}}(x) = \boldsymbol{A_2}\sigma(\boldsymbol{A_1 X} + \boldsymbol{b_1}) + \boldsymbol{b_2}$

# Universal approximators (iii)

**Park et al. - spiking neural networks**

**Park et al. - spiking neural networks**

# Dynamical systems

## State space models

System of differential equations of one variables of the form

$$\dot{x} = Ax + Bu, \quad y = Cx, \quad x_0 = x|_{t=0}$$

## Non-linear - State space models

System of differential equations of one variables of the form

$$\dot{x} = A(x)x + B(x)u, \quad y = C(x)x, \quad x_0 := x|_{t=0}$$

# Dynamical systems (ii)

## Universal Differential Equations

System of differential equations of one variables of the form
$\dot{x} = \mathrm{NN}(\theta, x)$ note this is also known by forced stochastic delay partial differential equation(PDE) defined with embedded universal approximators.

# Deep learning models

**Multilayer perceptron - $\mathcal{U}$**

$\sum_{i=0}^{m} \sigma(A_i x + b_1)$

**Convolutional neural network - $\boldsymbol{f * g}$)**

$\int f * g$

## 1.d.a PINN methods

**Neural Ode**

**Attention(Data-Dependent Kernel (Nadaraya–Watson))**

# Deep learning models (ii)

# 2 Technical considerations

# Frameworks

Technical considerations in this project are made from first a purely standpoint which then is formulated in practical terms:

**theoretic**

- **mathematical concepts**:
  - automatic differentiation(for black-/graybox modelling)
  - ode solving (for gray-/whitebox modelling)
- **implementations of models/techniques**

# Frameworks (ii)

## Python

- **mathematical concepts**:
  - jax (AD)
  - equinox/diffrax(ODE solving)
- **implementations of models/techniques**
  - dynax (sysid/basisfitting)
  - jax sys-id (sysid)

# Frameworks (iii)

## Julia

- **mathematical concepts**:
  - DifferentiationInterface.jl (AD)
  - OrdinaryDiffEq.jl (ODE solving)
- **implementations of models/techniques**
  - ScimlSensitivity.jl (sysid/nn fitting)
  - DatadrivenDiffEq.jl (Sindy)

# simplification of the Nonlinear problem

To avoid making the ode too stiff, i think of deploying https://juliapackages.com/p/varpro. Varpro is based on theoretical findings that any NL with a linear part can be recast to a purely nonlinear problem

# simplification of the Nonlinear problem (ii)


- [1] Golub, G.H., Pereyra, V.: “The differentiation of pseudoinverses and nonlinear least squares problems whose variables separate”. SIAM Journal on Numerical Analysis 10, pp 413-432 (1973)
- [2] Golub, G.H., Pereyra, V.: “Separable nonlinear least squares: The variable projection method and its applications”. Inverse Problems 19 (2), R1–R26 (2003)
- [3] Pereyra, V., Scherer, eds: “Exponential Data Fitting and its Applications” Bentham Books, ISBN: 978-1-60805-048-2 (2010)
- [4] Dianne P. O’Leary, Bert W. Rust: “Variable projection for nonlinear least squares problems”. Computational Optimization and Applications April 2013, Volume 54, Issue 3, pp 579-593 Available here
- [5] B. P. Abbott el. al. “ASTROPHYSICAL IMPLICATIONS OF THE BINARY BLACK HOLE MERGER GW150914” The Astrophysical Journal Letters, Volume 818, Number 2
- [6] J.E. Dennis, D.M. Gay, R.E. Welsch, “An Adaptive Nonlinear Least-Squares Algorithm”, ACM Transactions on Mathematical Software (TOMS), Volume 7 Issue 3, Sept. 1981, pp 348-368, ACM New York, NY, USA see here

## simplification of the Nonlinear problem (iii)

- [7] J.E. Dennis, D.M. Gay, R.E. Welsch, “Algorithm 573: NL2SOL—An Adaptive Nonlinear Least-Squares Algorithm”, ACM Transactions on Mathematical Software (TOMS), Volume 7 Issue 3, Sept. 1981, pp 369-383, ACM New York, NY, USA

# 3 Pre-Week meeting #1

# Motivation

Estimation of linear parameters (e.g., Re, Le, Bl, Mm, Km, Rm) in our loudspeaker dynamic model is crucial for:

- **Accurate System Identification**: Capturing the true electrical and mechanical dynamics ensures that controller design and simulation reflect real-world behavior.
- **Computational Efficiency**: VarPro eliminates the need to estimate linear coefficients in each nonlinear iteration, yielding faster and more stable parameter convergence.

## 3.b.a Estimation of linear params

In a linear time-invariant loudspeaker model

$$\dot{u} = A(\theta, u) + B(x)$$
$$y = C(u)$$

with parameter vector

$$\theta = [R_e, L_e, Bl, M_m, K_m, R_m]$$

and Gaussian measurement noise, the maximum-likelihood estimate is the weighted least-squares solution:

$$\hat{\theta} = \arg\min_{\theta} \sum_{n=1}^{N} \left[ \frac{(i[n] - \hat{\imath}[n;\theta])^2}{\sigma_i^2} + \frac{(v[n] - \hat{v}[n;\theta])^2}{\sigma_v^2} \right].$$

# Theory – Linear Parameter Estimation (ii)

Using **Variable Projection (VarPro)**:

- Separate linear-in-$\theta$ blocks (impulse responses) from nonlinear parameters.
- Project out the linear coefficients analytically at each iteration.
- Solve a lower-dimensional nonlinear least-squares on $\theta$.

# Hypothesis

1. **Time-domain WLS** recovers $\theta$ with accuracy comparable to cross-spectral methods under 5% noise.
2. **Welch-based TF-LS** exhibits higher variance and bias at high frequencies due to window leakage.
3. **Multitaper + Wiener** yields the lowest PSD-estimation variance but at greater computational cost.

# Planned Experiments

1. **Data generation**
   - Simulate true model with known $\theta_{\text{true}}$.
   - Drive with (a) unit impulse, (b) white noise; sample $N = 5 * 10^6$ points, add 5% uniform noise.
2. **Parameter estimation methods**
   - **TD-WLS**: minimize $\sum(y - x(\theta))^2 \sigma^2$ via Levenberg–Marquardt.
   - **Welch TF-LS**: estimate $\hat{G}(\omega)$ with `DSP.welch`, fit $\sum| \hat{G} - G(\theta)|^2$.
   - **Cross-spectral WLS**: use weighted cross-spectral fit (Eq.18 Lab A).
3. **Evaluation metrics**
   - Parameter RMSE: $|\hat{}\theta - \theta_{\text{true}}|$.
   - FRF error: $\max_\omega |H_{\{\text{est}\}} - H_{\text{true}}|$.
   - Compute time per method.

# 4 Pre-Week meeting #2

# Research question

1. **Q:** What “surrogate model”/“In-situ-compensation” or “function approximation” minimizes the physics informed loss function of the chosen subset of models modelling the loudspeaker?
   - **Follow-up:** Whats the assumptions of the model? White-,Gray- or Black-box?
     - **A:** preferably Gray-box

- **Follow-up:** Which physics informed loss function?
  - **A:** MSE + PINN terms
- **Follow-up:** which model?
  - **potential models:**
    - polynomial basis - based on orthogonal basis
    - **Resovoir computing** - based on chaos
    - **neural networks** - based on linear combination of nonlinear basis
    - **kolmogorov arnold networks** - based on linear combination of nonlinear basis

2. **Q:** How can an analytical model be extracted from data, Black-box or Gray-box models?
   - **Follow-up:**

1