The difficulty in implementing the SINDY fit is that the basis functions are matrices and powers of matrices. There are at wo possible approaches:

1. Expand the basis matrix functions into their individual components. Since the basis functions are 3x3 and are symmetric matrices.

2. Somehow treat matrices as matrices. Come up with a algorithm that operates on matrices. This would require developing new algorithm. Perhaps I should understand how RUDE works.

The first would be to express the basis functions assuming that σ_{12} and $\sigma 23$ are zero.

In Sachin's project, are we seeking an analytical expression of the unknown $F(\cdots)$ that is parametrized by λ, G, etc? Or are we fitting the term for each set of parameter and initial condition values (which is easier, but still hard.)

1 Sparse Linear Regression of 1-D function

Consider the function $g(u)$ given as $g_i = g(u_i)$, $i = [0, 1, \cdots, N]$:

$$g(u) = \sum_j a_j \phi_j(u)$$

where $\phi_j(x)$ are scalar basis functions. The key is that we wish to do sparse regression. How is this done? Maybe with L_1 normalization. We have that

$$g(u_i) = \sum_j a_j \phi_j(u_i)$$

We will minimize

$$\mathcal{L} = \sum_i (g(u_i) - g^*(u_i))^2 = \mathcal{L}(a_1, \cdots, a_N)$$

where g^*_i is synthetic or experimental data, mostly equally-spaced. Problably should add a L_1 norm $\lambda_1 \sum_i |a_i|$. How to choose the Lagrange multiplier λ_i?

We can use SGD, ADAM, BFGS to minimize a loss function between the values g_i and $f_i = f(u_i)$.

2 Sparse linear regression of a matrix equation

Consider the matrix function $G(u)$ given as $G_i = G(U_i)$, $i = [0, 1, \cdots, N]$:

$$G(u) = \sum_j a_j \Phi_j(u)$$

where $\Phi_j(x)$ are matrix basis functions. The key is that we wish to do sparse regression. How is this done? Maybe with L_1 normalization. We have that

$$G(u_i) = \sum_j a_j \Phi_j(u_i)$$

where a_j are scalar coefficients. We will minimize

$$\mathcal{L} = \sum_{i=1}^{N} ||G(u_i) - G^*(u_i)||^2 = \mathcal{L}(a_1, \cdots, a_N)$$

where N is the number of points, and M is the number of basis functions, and where G_i^* is synthetic or experimental data, mostly equally-spaced. Each norm has the form:

$$N_i = ||\sum_j a_j \Phi_j(u_i) - G_i^*||^2 \quad (1)$$

$$= ||\sum_j a_j \Phi_j(u_i) - G_i^*||^2 \quad (2)$$

where Φ_j and G_i^* are 3×3 matrices.

Problably should add a L_1 norm $\lambda_1 \sum_i |a_i|$. How to choose the Lagrange multiplier λ_i?

We can use SGD, ADAM, BFGS to minimize a loss function between the values g_i and $f_i = f(u_i)$.