Let $x(t;p) \in R^d$, typically $d = O(10)$, t is the time and $p \in R^{n_p}$ is a parameter vector.

x is given by the solution of the following ODE:

$$\frac{dx}{dt} = f(x,p); \qquad x(0) = g, \tag{1}$$

where g is the initial condition. The solution $x(t)$ is parameterized by g and p.

We want to solve the follwing problem: Given $\{p_i, g_i\}_{i=1}^{s}$, solve eq. (1) using a GPU accelerated solver that solves simultaneously eq. (1) for all i.

On a CPU this can be done by assigning an instance of eq. (1) to each thread, and that would be a good way to start as a benchmarking and check correctness, since each thread can invoke an existing ODE solver.

On a GPU it will more involved because GPUs require vectorization, massive multithreading and essentially a single logic running across all threads.

Challenges include the following:

1. The stiffness characteristics of the solver depend on the parameters p and perhaps the initial condition g. As a result either an explicit or implicit solver will have variable time step dt_i for different values of i.
2. As a result different instances may terminate at different times.

The project would have to develop a novel adaptive solver that supports both explicit and implicit solvers with adaptive time stepping. It would involve the following steps:

1. Implement on CPU and multithreading to get correctness.
2. Implement simple adaptive RK45 and BDF methods on CPU to check correctness
3. Implement nonadaptive RK45 on GPU for non-stiff models for batched using variable dt for each i
4. Implemennt adaptive RK45 on GPU for non-stiff models
5. think about implicit solver using jacobian and matrix free jacobian

Goal is to have massive speed ups on GPU. The implicit case is the most interesting.

Suggeston for implicit: batched inverse for the jacobian calculation; batched jacobian calculations assuming quadratic RHS implimeneted (other wise user defined); if inversion to expensive consider preconditioned GMRES as a decoupled system; periodically balance ODEs that are done. Automatically deal with larger ODEs