Graphical processing units (GPUs) are massively parallel computing devices whose use has recently been explored in numerical weather prediction (NWP). Their parallel floating-point capabilities match well with the inherently parallel nature of NWP calculations and can result in significant computational speed-up. A GPU implementation of the ARW dynamical solver will be presented.

The goal of the implementation is twofold: The computation should run entirely on the GPU, with the sole role of the central processing unit (CPU) being to manage the execution of the GPU kernels. That is, no part of the dynamical solver is to be executed on the CPU. The advantage of this is that all model data then resides exclusively on the GPU for the duration of the model run. This ensures that data does not need to be transferred back and forth between the CPU and GPU at each timestep, which would otherwise occur if some parts of the model execute on the CPU and others on the GPU. With current GPU hardware, data transfer to and from the GPU often becomes a bottleneck and can therefore limit the speedup attained with a GPU implementation. Secondly, the implementation should allow for a direct comparison of running the code entirely on the CPU and entirely on the GPU without writing separate code for each type of hardware. Thus a single source code tree should be used for both CPU and GPU versions. This requirement is realizable because of the similarity between the GPU programming language CUDA and ANSI C, the latter of which is used when the simulation runs entirely on the CPU.

The programming challenges involved with this implementation as well as preliminary speed-up results are presented.