

# Automatic GPU Code Generation of Modelica Functions

Hilding Elmqvist<sup>1</sup> Hans Olsson<sup>1</sup> Axel Goteman<sup>1,2</sup> Wilhelm Roxling<sup>1,2</sup>  
Dirk Zimmer<sup>3</sup> Alexander Pollok<sup>3</sup>

<sup>1</sup>Dassault Systemes AB, Lund, Sweden,  
{Hilding.Elmqvist, Hans.Olsson}@3ds.com

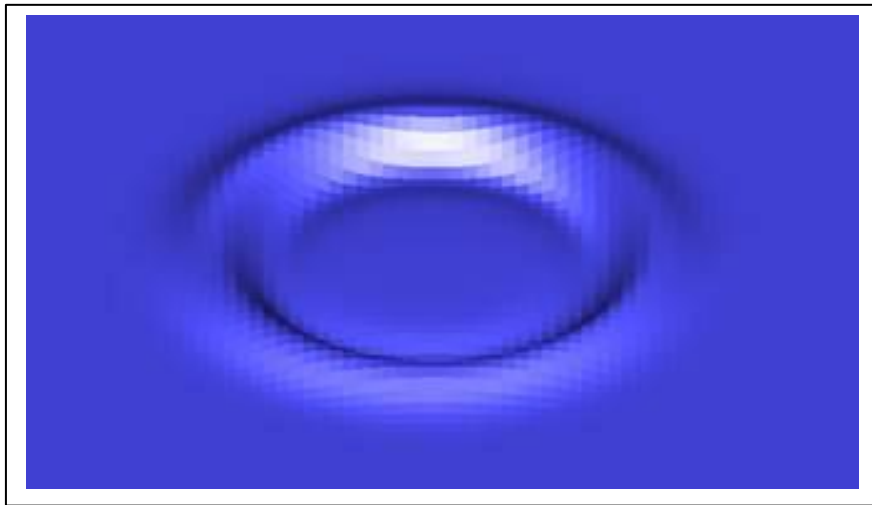
<sup>2</sup>Lund Institute of Technology, Lund, Sweden,  
{axel.goteman, wilhelm.roxling}@gmail.com

<sup>3</sup>Institute of System Dynamics and Control, DLR, Germany,  
{Dirk.Zimmer, Alexander.Pollok}@dlr.de

Modelica users can and want to build more realistic and complex models. This typically means slower simulations. In the past, the speed of single CPUs has increased significantly to partly compensate, but more recently, there has been a shift to multi-core architectures. This is taken to the extreme in Graphics Processing Units (GPUs).

This paper discusses code generation for GPU cores. This is important when the model has regular structure, for example, discretization of PDEs. The behavior of each cell can then be partly described by a function call. The evaluation of such calls can then be made in parallel on the GPU cores. The same function is thus executed on every GPU core, but operates on different data; the data of its cell.

Our GPU code generator automatically generates code for Modelica functions, i.e. no additional language constructs are needed. The function is just annotated as suitable for execution on a GPU.



**Figure 1.** Dymola animation of a circular wave front, calculated by a GPU