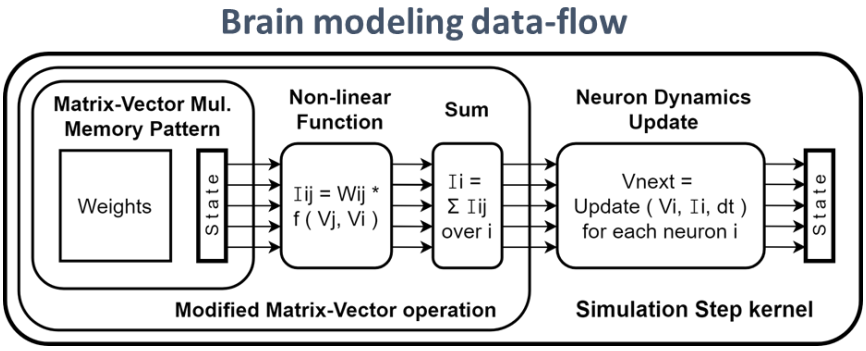
EXA2PRO SkePU tool applied to a brain modeling application

A. Short Application description:

The brain modeling application simulates a biologically-detailed neural network. It can be thought of as a large electrical circuit. The non-linear dynamics require step-by-step simulation and the continuous interaction between neurons requires communication at each step.



The specific application simulates a cluster of neurons.

At each simulation step, for each neuron the following are applied:

- Summation of electrical current from adjacent neurons
- Simulation progresses based on internal dynamics

B. Applying EXA2PRO framework to brain modeling (SkePU tool):

The Map skeleton uses an input collection, to produce an output element for each one in the input collection, using a user-written function. Therefore, the **Map skeleton** applies.

We developed a function that produces each neuron's updated state, using present data and the weight matrix, which was used by the Map skeleton.

Function signature:

```
OutElm mapFunc( skepu::Index idx, InElm in1, InElm in2 ..., SharedVar shared1, ... )
```

Usage:

```
auto RunnableKernel = Map(mapFunc)
skepu::Vector<InElm> inVec1, inVec2; SharedVar sharedVar; skepu::Vector<OutElm> outVec;
RunnableKernel( outVec, inVec1, inVec2, sharedVar );
```

Original code:

```
for each timestep:
  for each neuron i:
    Get Isyn from adj. Neurons
    … internal dynamics code …
    Compute stateNext[i]
```

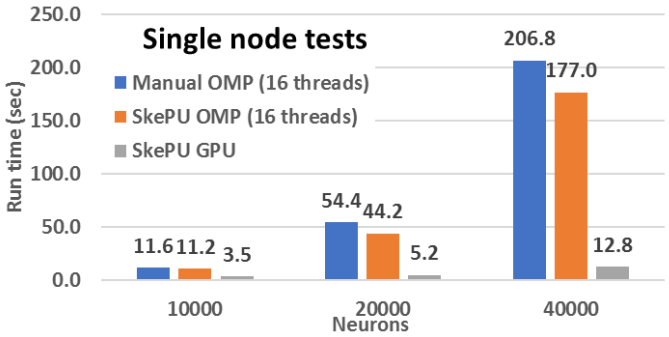
SkePUized code:

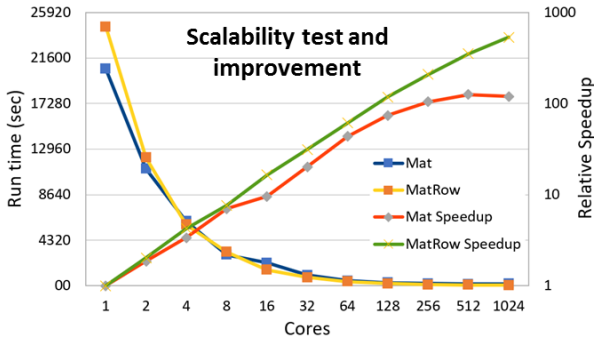
User function

```
Function get_stateNext(i, stateNow, adj.matrix):
  Get Isyn from adj. Neurons
  … internal dynamics code …
  Return stateNext for neuron I

Kernel = Map(get_stateNext)
for each timestep:
  stateNext = Kernel(stateNow, adj.matrix)
```

C. Results





Single node tests were conducted on a 2x Intel Xeon Gold 6138, 2x20 H/T cores with 128 GB RAM that integrates an nVidia Tesla V100 GPU with 32GB VRAM. We noticed that applying the EXA2PRO framework, and more specifically SkePU, results in efficient use of parallel hardware, including GPU.

The scalability test of brain modeling was conducted in the Tetralith cluster, using 32 nodes with 2x Intel Xeon Gold 6130, 2x16 H/T cores and 96 GB RAM per node. Initially, scalability issues were identified (red line). However, by optimizing the data container scalability of brain modeling was significantly improved (green line).

D. Conclusions

Applying the EXA2PRO framework to the brain modeling applications, enabled the porting of the application to GPU accelerators and cluster resources. The results, in terms of performance were efficient and limited changes to the source code were required.

More Information: S. Panagiotou, A. Ernstsson, J. Ahlqvist, L. Papadopoulos, C. Kessler, D. Soudris," Portable exploitation of parallel and heterogeneous HPC architectures in neural simulation using SkePU", Proceedings of the 23th International Workshop on Software and Compilers for Embedded Systems, 2020.