Performance and Scaling for uG4

Our base code uG4 is parallelized using MPI and written in C++ and has been extensively benchmarked elsewhere (Vogel et al., 2013; Heppner et al., 2013) and shown to scale to $> 200{,}000$ processes. The solver interfaces of uG4 are designed with parallel computing in mind. The first benchmark or standard test problem is the Laplace equation in three dimensional space on a unit cube. The second order elliptic partial differential equation reads, where Φ is a scalar function in the domain $\Omega \subset \mathbf{R}^n$, and Δ is the Laplace operator:

$$\Delta\Phi = 0 \tag{1}$$

Strong scaling is demonstrated for a sevenfold regularly refined unit cube, cf. Tab. 1 and Fig. 1. The execution time roughly halves by doubling the number of processes. A weak scaling study, cf. Tab. 2 and Fig. 2, 3 on SDSC Comet and TACC Stampede2 confirms the scaling behavior. With each refinement of the grid the degrees of freedom (DoFs) increase by a factor of 8 and thus we increased the number of processes eightfold in each refinement step.

Strong scaling for first problem in uG4			
DoFs	*Runtime [s]*	*# Processes*	*# Nodes*
16,974,593	41	24	1
16,974,593	20	48	2
16,974,593	14	72	3
16,974,593	10	96	4
16,974,593	8	120	5
16,974,593	7	144	6
16,974,593	5	168	7
16,974,593	2	192	8

Table 1: Runtimes of simulations of a **3d Laplace problem** on a fixed unit cube domain (seven regular refinements) with uG4 on the SDSC Comet HPC cluster. Note that each node can allocate at maximum 24 processes and the maximum number of processes or cores per job is limited to 1728 processes in total.

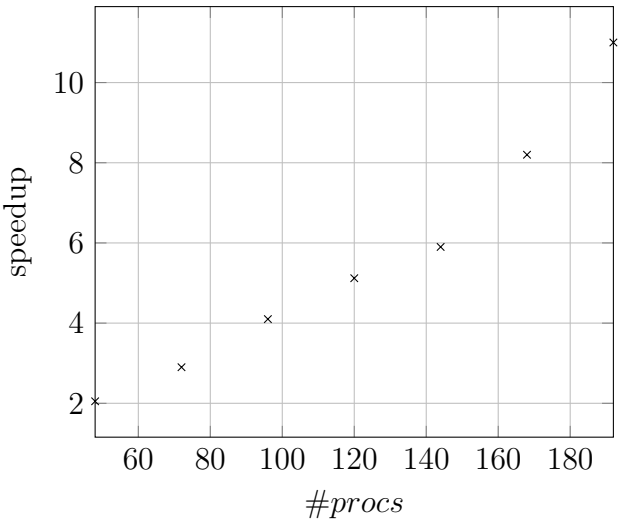

Figure 1: Speedup of distributed execution. #procs denotes the involved processes on the SDSC system Comet.

Weak scaling for first problem in uG4			
DoFs	*Runtime [s]*	*# Processes*	*# Grid refinements*
27	5.06	1	0
216	5.06	8	1
1,726	6.92	64	2
13,824	6.92	512	3
110,592	5.50	4096	4

Table 2: Runtimes of simulations of a **3d Laplace problem** on a unit cube domain (different number of regular refinements) with uG4 on the SDSC Comet HPC cluster. Note that each grid refinement increases the DoFs by a factor 8.

Weak scaling for first problem in uG4			
DoFs	*Total runtime [s]*	*# Processes*	*# Grid refinements*
27	1	1	0
216	1	8	1
1,726	1	64	2
13,824	1	512	3
110,592	1	4096	4
884,736	2	32768	5

Table 3: Runtimes of simulations of a **3d Laplace problem** on a unit cube domain (different number of regular refinements) with uG4 on the TACC Stampede2 HPC cluster. Note that each grid refinement increases the DoFs by a factor of 8. Note that the **large** queue on Stampede2 is not yet available to us to benchmark, thus we are restricted to process count up to 32768.

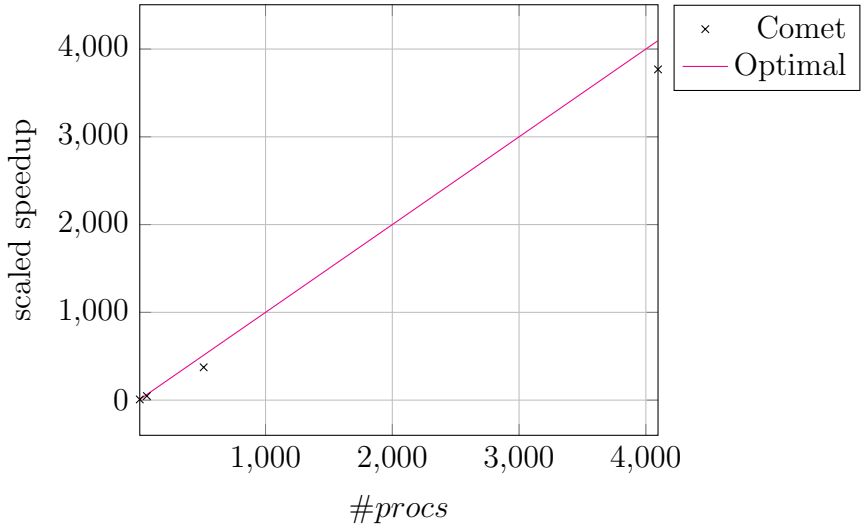

Figure 2: Scaled speedup of distributed execution. #procs denotes the involved processes on the SDSC system Comet. Note the optimal scaling and achieved scaling behaviour of the test problem.

The second problem is motivated by neurobiology. Of interest are the three-dimensional spatio-temporal Ca^{2+} and IP_3 dynamics in the intracellular space of a neuron respectively spine. This is modeled by a (system) of diffusion-reaction equations, cf. Eqns. 3-7 in proposal.

$$\frac{\partial u}{\partial t} = \nabla \cdot (D \nabla u) \tag{2}$$

where $u(x,t)$ stands for one of the quantities above.
The full domain equations for cytosolic calcium and calbindin are thus given by

$$\frac{\partial c_c}{\partial t} = \nabla \cdot (D_c \nabla c_c) + \left(\kappa_b^- \left(b^{\text{tot}} - b\right) - \kappa_b^+ \, b \, c_c\right), \tag{3}$$

$$\frac{\partial b}{\partial t} = \nabla \cdot (D_b \nabla b) + \left(\kappa_b^- \left(b^{\text{tot}} - b\right) - \kappa_b^+ \, b \, c_c\right) \tag{4}$$

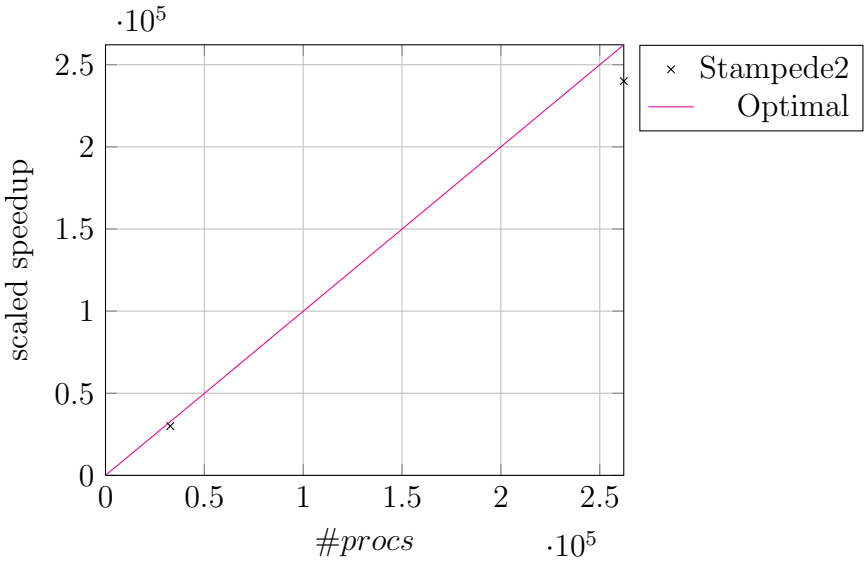

Figure 3: Scaled speedup of distributed execution. #procs denotes the involved processes on the TACC system Stampede2. Note the optimal scaling and achieved scaling behaviour of the test problem.

The computational grid is the reconstruction of a synaptic spine in three-dimensional space as depicted in the accompanying proposal document. Starting on the base level with a large number of DoFs the grid is refined three times. The cost for solving the problem increases only slightly but remains bound up to the testable limit, cf. scaling in Fig. 4 and Tab. 4.

Weak scaling for second problem in uG4			
DoFs	*Runtime [s]*	*# Processes*	*# Grid refinements*
547,348	37.1	24	0
4,378,784	44	192	1
35,030,272	57	1536	2

Table 4: Runtime of a simulations on a **spine** reconstruction in three-dimensional space. Note the grid has been regularly refined two times with uG4 and the DoFs increase by a factor 8 and thus also the processes have to increase eightfold. Runtime cost increases slightly but remains bound.

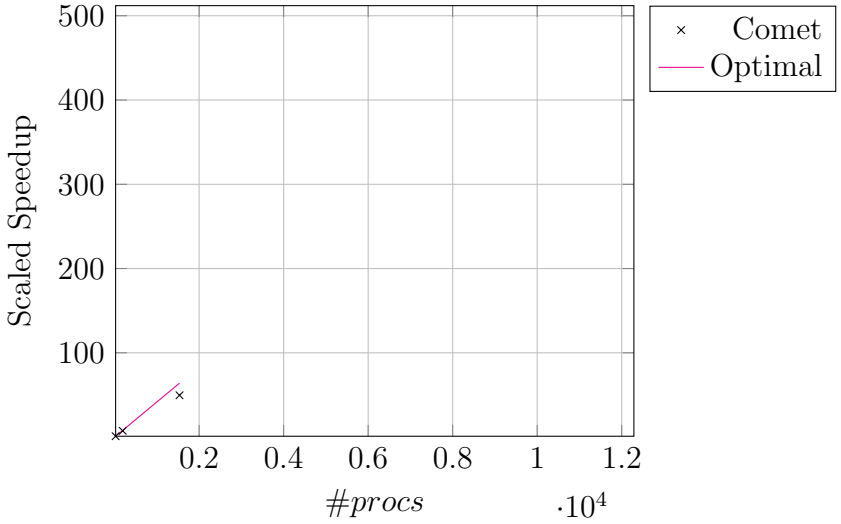

Figure 4: Scaled speedup of distributed execution. #procs denotes the involved processes on the SDSC system Comet.