Fraunhofer

SCAI

FRAUNHOFER INSTITUTE FOR ALGORITHMS AND SCIENTIFIC COMPUTING SCAI

FAST AUTO-PARALLEL LINEAR SOLVER FOR HUGE ENGINEERING APPLICATIONS

Fraunhofer Institute for Algorithms and Scientific Computing SCAI

Schloss Birlinghoven 1
53757 Sankt Augustin
Germany

Contact:

Dr. Hans-Joachim Plum
Phone +49 2241 14-4034
samg@scai.fraunhofer.de

www.scai.fraunhofer.de

XSAMG
Auto-parallel SAMG
for Clusters

The original SAMG is a very efficient linear solver library based on algebraic multigrid (AMG), specifically developed for industrial applications. SAMG supports both serial and multi-core computations on single PC, workstations or compute nodes.
The new XSAMG library additionally exploits the parallelism offered by multiple nodes of a compute cluster by distributing linear systems across different nodes (based on MPI). Since this is done automatically, the calling simulation program itself does not have to be prepared for distributed computing.

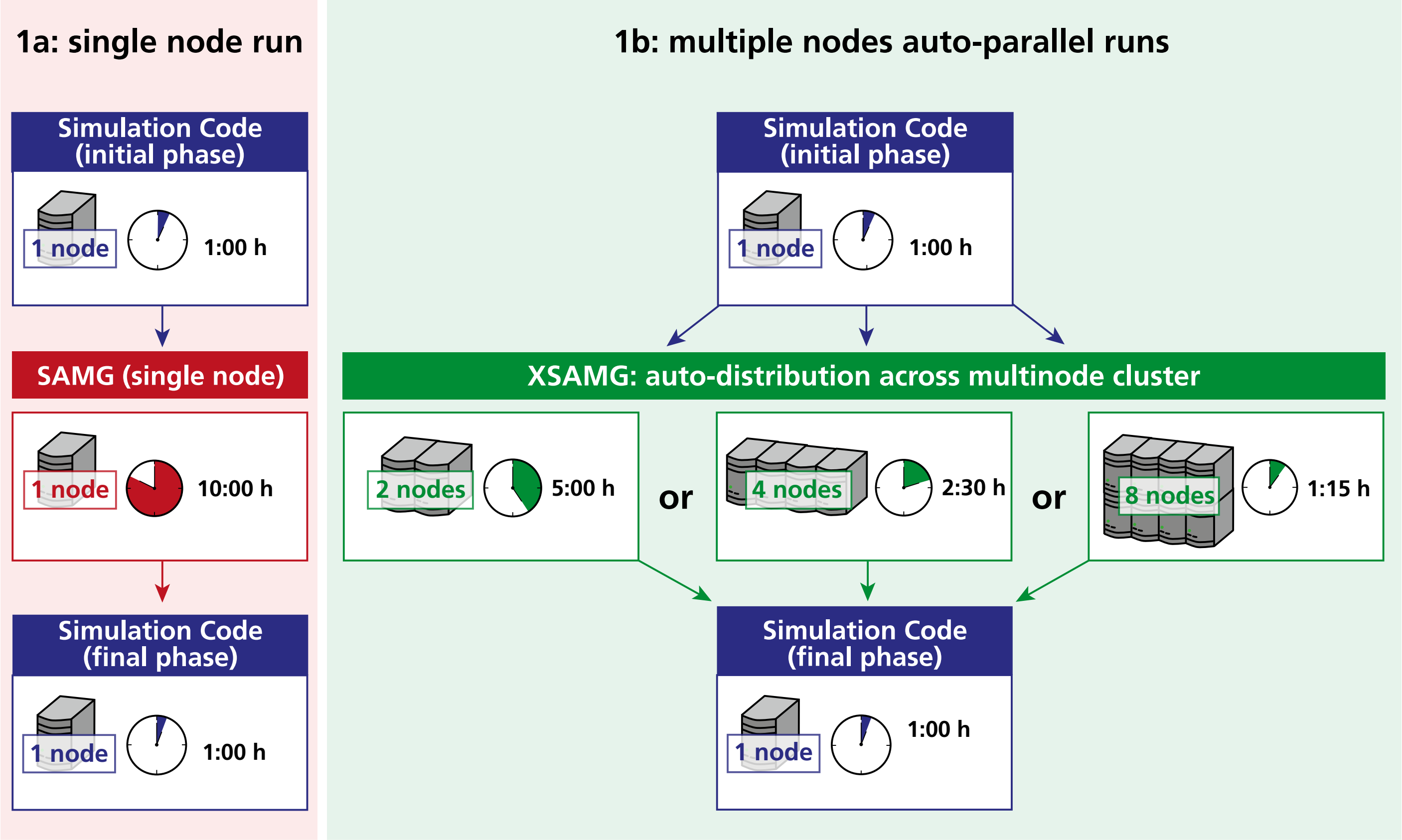
XSAMG makes the linear solver part of a standard code “cluster aware” – without re-engineering

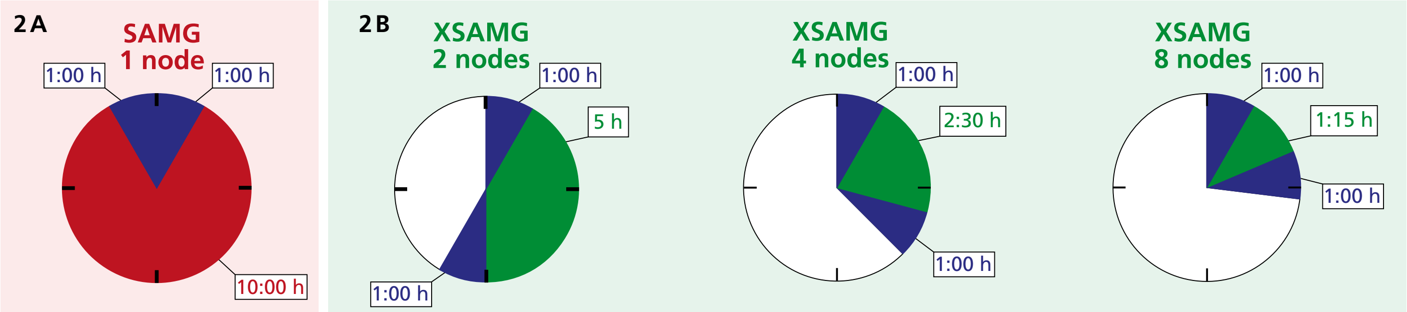
That is, XSAMG lifts a user’s single-node application beyond the performance limits of a single node by making the linear solver phase “cluster aware”, this way drastically reducing solution times further without the need of a user to be concerned about the complicated underlying parallel cluster software infrastructure.

XSAMG offers the standard way of exploiting multiple cores (“multi-threading”), plus the usage of multiple compute cluster nodes.

XSAMG – enhanced parallelism, enhanced flexibility

Thus, it extends over all architectural components of a modern parallel architecture and lets the user easily select the best option for her/his code.


1a: single node run
Simulation Code (initial phase)
1 node
1:00 h
SAMG (single node)
1 node
10:00 h
Simulation Code (final phase)
1 node
1:00 h
1b: multiple nodes auto-parallel runs
Simulation Code (initial phase)
1 node
1:00 h
XSAMG: auto-distribution across multinode cluster
2 nodes
5:00 h
or
4 nodes
2:30 h
or
8 nodes
1:15 h
Simulation Code (final phase)
1 node
1:00 h



1 A *SAMG on 1 node*

1 B *XSAMG exploiting 2, 4 or 8 nodes*

2 A *Single-node multi-core use of SAMG: total run time (12h)*

2 B *Multi-node multi-core use of XSAMG: Ideal total time for increasing number of nodes*

Who should consider using XSAMG?

Suppose that a simulation code calls, as a core section, a linear solver and that the code developer already uses (or plans to use) SAMG. Then, using XSAMG instead is a favorable option if the following conditions are met:

- The overall simulation code is limited to single nodes (either serial or OpenMP parallel).
- The code developer does not want to invest the effort to make the simulation code MPI-parallel.
- The solution of linear systems requires a significant part of the overall simulation time, even with the very efficient SAMG.
- The code developer and/or the customers have access to a cluster computer.

How does it work?

XAMG is based on the following flexible execution model: One “head” node performs the user’s simulation tasks, the SAMG solver is automatically spread across additional remote nodes – the more nodes deployed, the faster the overall execution speed (cf. Fig 1a and 1b). Note that XSAMG automatically exploits the parallelism offered by both multiple cores and multiple nodes, even if the user’s single-node program does not exploit multi-core computing at all.

What is the potential benefit?

Suppose a single-node application uses (serial or OpenMP parallel) SAMG as a linear solver. Just as an example, let us assume that the complete application needs 12h from which the linear solver takes up 10h (shown in red, cf. Fig 2a). Fig 2b shows how, ideally, the run time of the overall application could benefit from the cluster readiness of XSAMG: In principle, the pure solution time (shown in green) gets smaller as more nodes are employed. In-line with Amdahl’s law, the asymptotic minimum time required in this example is 2h.