

Scalable Resolution of NP-hard Problems (DE/EN)

Within the broad area of resolving combinatorial NP-hard problems in a scalable manner, changing topics are available on request in the following areas:

High Performance SAT Solving

The SAT solving engine which is part of our system **Mallob** is currently the best performing large-scale SAT solving system in the world. There are still many open tasks concerning our system, such as finding a way to select a solver portfolio dynamically on a per-instance basis, or exploring more promising approaches to diversify individual solvers.

Malleable Scheduling of NP-hard Applications

Mallob cannot only solve one propositional formula at a time, but offers a generic interface to solve several problems at once. These tasks are then balanced among all processing units in a fair manner using decentralized distributed algorithms. For this purpose, we design our tasks to be *malleable*, i.e., to support a fluctuating number of compute nodes during the calculation. In addition to SAT Solving, malleable interfaces for further applications can be explored, such as automated planning or SMT solving.

Hierarchical Automated Planning

Our planning system **Lilotane** solves hierarchical planning problems by reducing them to propositional logic without performing an expensive preprocessing previous approaches rely on. As such Lilotane scored the 2nd place in the Total Order Track of the International Planning Competition 2020. Several aspects of this approach can be improved and extended, such as finding optimal plans, extending the underlying planning model, or researching alternative SAT encodings.

Requirements and Organization

For most of the topics solid programming skills in C++ are required (depending on the project, these skills can be acquired over the course of the project). For topics with parallel / distributed programming, basic knowledge with MPI is useful.

A thesis can be supervised in German or English and should be written in English. If interested, please contact Dominik Schreiber <dominik.schreiber@kit.edu>.

References

Mallob : <https://github.com/domschrei/mallob>

Lilotane : <https://github.com/domschrei/lilotane>