Plan Overview

A Data Management Plan created using DMPonline.be

Title: Certified First-Order Model Expansion (CertiFOX)

Creator: Bart Bogaerts

Principal Investigator: Bart Bogaerts

Affiliation: KU Leuven (KUL)

Funder: European Research Council (ERC)

Template: ERC DMP +

Principal Investigator: Bart Bogaerts

Project abstract:

The field of combinatorial optimization is concerned with developing generic tools that take a declarative problem description and automatically compute an optimal solution to it. Often, users specify their problem in a high-level, human-understandable formal language. This specification is first translated into a low-level specification a solver understands and subsequently solved. Thanks to tremendous progress in solving technology, we can now solve a wide variety of NP-hard (or worse) problems in practice. Moreover, these tools are increasingly used in real-life applications, including high-value and life-affecting decisions. Therefore, it is of utmost importance that they be completely reliable. The central objective of this proposal is to develop methodologies and tools with which we can guarantee with 100% certainty that the right problem has been solved correctly.

To achieve this ambitious objective, we will build on recent breakthroughs in proof logging, where solvers do not just output an answer, but also a proof (or certificate) of correctness. However, a major limitation of current techniques is that correctness is not proven relative to the human-understandable specification written by the user, but relative to the low-level translation that the solver receives, meaning that there is no guarantee that the solver is solving the original problem. In this project, we will investigate end-to end guarantees of correctness. When successful, this will have a major impact on the way combinatorial optimization software is developed, evaluated, and used: the proofs produced will enable (1) debugging, since proofs contain detailed information about where bugs occurred, (2) auditability, since proofs can be stored and checked by an independent third party, and even (3) rigorous evaluation of algorithmic improvements. There are several fundamental questions that need to be tackled before we get there, relating to the nature of proofs for grounding, proofs of transformations, and the combination of proofs.

ID: 211104

Start date: 01-10-2024

End date: 30-09-2029

Last modified: 20-11-2024

Certified First-Order Model Expansion (CertiFOX)
GDPR Record

GDPR record

Have you registered personal data processing activities for this project?

- No

We do not process any personal data

Certified First-Order Model Expansion (CertiFOX)
DPIA

DPIA

Have you performed a DPIA for the personal data processing activities for this project?

- Not applicable

We do not process any personal data

Certified First-Order Model Expansion (CertiFOX)
ERC DMP +

Project information

Project Acronym

CertiFOX

Project Number

101122653

Data summary

Summary

We are researching fundamental algorithms for combinatorial optimization. Most research will be done on publicly available datasets without risk of abuse. No personal data will be collected for CertiFOX.

Overview of all the generated data:

Type	Description	New?	Formats	Size
Software	source code of proof checker and of proof-generating grounders and solvers.	Yes	Source files (.cpp, hpp, py) and documentation of code (.md, .txt)	<1 Gb
Instances	Benchmarking instances, for testing proof generating methods on	No: Re-used from (future or past) solving competitions, eg https://www.minizinc.org/challenge.html and https://satcompetition.github.io/ and https://maxsat-evaluations.github.io/2024/ Yes: if needed more (synthetic) data will be generated.	.cnf, .wcnf, .asp, .lp, .mzn, .fzn, .idp	<100Gb
Proofs	generated by our tools during experimental evaluation	Yes	.pbp, .foxp	<1Tb
Scripts	Scripts for running, analyzing, and reproducing experiments	Yes	.sh, .py .r	< 1 Gb
Results	Experimental results (detailed statistics about runs, including time, memory usage, types of proof rules used)	Yes	.txt .csv	<10Gb
Papers	Source files for papers and images (eg graphs summarizing runtimes) generated from the experiments	Yes	.tex .jpg, .png	<1Gb

This data will be managed on a per paper basis.
The general principle is that:

- Before acceptance of a paper, all data will be closed, but accessible by project members and safely backed up
- After acceptance of a paper, all relevant data will be open.

FAIR data

1. Making data findable

Upon acceptance of a paper:

- For all the **software** developed for that paper:
  - The code will be made available on a public git server (e.g., gitlab)
  - A snapshot will be published on the KU Leuven RDR repository (trusted data repository with DOI) or Zenodo (including a commit hash linking it to the aforementioned git repository)
- All **instances** used will be published on the KU Leuven RDR repository
- All **scripts** needed to reproduce the experiments will be published on the KU Leuven RDR or Zenodo repository
- The author accepted manuscript (**pdf file)** of the paper will be deposited in Lirias (KU Leuven's trusted paper repository)

We will use the metadata of these repositories to ensure all data is findable and linked.

2. Making data openly accessible

Upon acceptance of a paper:, Software, scripts, instances, and pdf files (paper) will be made openly accessible through KU Leuven's trusted repositories.
The actual proofs and detailed run results will not be made open since they can be reproduced (and since proofs can get very large).

Next to storing this snapshot in a data repository (Zenodo or KU Leuven's RDR) and a versioned history on a git repository, we ensure maximal openness and follow best practices in our research domain.

3. Making data interoperable

We use standard formats for all data (for instances instances for SAT solvers will be stored as dimacs .cnf files, as common in the field). The only exception is the proofs, which will be in a new format, with provisionary name "First Order model eXpansion proof .foxp".

4. Increase data re-use

The instances will be released under a CC-BY license.
The source code will be released under a Apache-2.0 or MIT License.

These will be made available at the latest on the publication time of the corresponding paper.

The data will remain reusable at least for the duration of the hosting service that is hosting it (zenodo, github, KU Leuven RDR) as well as on our longterm snapshot storage (guaranteed at least 10 years).

5. Allocation of resources and data security

Overview of where data will be stored.

Type	Before publication	After publication
Software	KU Leuven gitlab server or Private github	Public github **and** (KU Leuven RDR or Zenodo)
Instances	KU Leuven gitlab server or Private github	KU Leuven RDR or Zenodo
Proofs	Departmental storage	Departmental storage (not public: reproducible)
Scripts	KU Leuven gitlab server or Private github	KU Leuven RDR or Zenodo
Results	KU Leuven gitlab server or Private github	KU Leuven gitlab server (not public: reproducible)
Papers	KU Leuven gitlab server or Private github	PDF files: in Lirias Source files: KU Leuven gitlab server

The KU Leuven data policy requires retention of relevant data for a period of at least 10 years after the end of a project (barring 3rd party agreements).
KU Leuven supports its researchers in the area of RDM (research data management) by the provision of a customized data management plan (DMP) (including guidance to KU Leuven legal requirements and policy guidelines) based on the templates provided by the Digital Curation Centre (UK), and a dedicated RDM support desk advising on data storage, metadata and preparing data for sharing. The university also continues to invest in affordable long-term storage and curation.

This project contributes to the facilities owned and operated by the research section and department. This includes long-term, secure data storage (NetApp), Git repository servers, web servers and backup facilities. This amounts to approximately 2500 EUR / year.

The PI will be responsible for implementing the DMP, and the host institute (KU Leuven) offers support for data management. The PI will update the DMP anytime conditions change. A midterm review will be accompanied by a detailed DMP and a final reviewed DMP will be sent along with the final report.