MIP 2022 Computational Competition

The computational development of optimization tools is a key component within the MIP community and has proven to be a very challenging task: it requires great knowledge of the well-established methods, technical implementation abilities, as well as creativity to push the boundaries with novel solutions. The main goal of the MIP 2022 Computational Competition is to encourage and provide recognition to the development of novel practical techniques within MIP technology. In its first edition, the computational competition will focus on finding good-quality solutions to general **Mixed-Integer Linear Optimization problems**.

The challenge

Participants of this competition must **create a novel general-purpose primal heuristic** for mixed-integer linear optimization problems. Participants are encouraged to develop **LP-free heuristics**, however, everyone is allowed to solve auxiliary **convex optimization problems**. High-quality submissions will be offered an expedited review process in Mathematical Programming Computation. See more details on the rules below.

Part of the instances will be drawn from MIPLIB 2017 (https://miplib.zib.de/) and the rest will come from undisclosed sources.

1 Rules

1.1 General code rules

- The use of existing general-purpose optimization software **at runtime** (CPLEX, Gurobi, Xpress, SCIP, etc) is allowed, but limited to only solve **purely continuous convex optimization problems**. Branching is allowed, but only if coded by the participants.
- Offline solving of MIPs is allowed, for example, for training phases of learning-based methods.
- The use of a solver's **pre-solve** capability is also allowed.
- The source code may be written in any programming language and participants should be prepared to compile and run their code on a Linux server.
- A solution will be considered feasible if the absolute violation of every constraint is at most 10^{-5} and if all integer variables are at most 10^{-4} away from the nearest integer value. That is, for each constraint $a^\top x \leq b$, a solution $\hat{x}$ should satisfy

  $$a^\top \hat{x} - b \leq 10^{-5},$$

  and if x_i is an integer variable, it should hold that $|\hat{x}_i - [\hat{x}_i]| \leq 10^{-4}$, where $[\cdot]$ rounds to the nearest integer.
- All reported results must use **no more than 8 threads and 16 GB of RAM**, with a time limit of **10 minutes** per instance.

1.2 Instances

The instances where the submissions will be tested are divided in two: **public and hidden**. The public instances (drawn from MIPLIB 2017) are available on the competition's website and must be reported in the authors' submission. In the second evaluation round (see below), the codes will be executed on the hidden instances. All instances will be in MPS file format.

1.3 Participants

There is no restriction on who can participate in this challenge, and the participation of students is highly encouraged. Besides the main award, the jury may decide to recognize outstanding student submissions: these will be submissions where the majority of the work was performed by students.

1.4 Evaluation procedure

1.4.1 First stage

All participants must submit a written report of **10 pages maximum** plus references, in Springer LNCS format, through the MIP 2022 website (`mixedinteger.org/2022`). Submissions will be accepted until **March 10th, 2022 at 8:00 PM EST**.

All reports must include the following information:

- Description of the methods developed and implemented, including the necessary citations.
- Computational results on all public instances, which should include at least the following two metrics for each instance: solution value (if feasible) and running time.

If the majority of the work was done by one or more students, the participants should include a **letter of attestation** indicating this.

Based on the reports, and using the evaluation criteria below, a jury will select **top submissions** to move to the second stage.

1.4.2 Second stage

In the second stage, all selected submissions will be given a user account on a Linux server, where participants will copy and compile their code (if necessary). The jury will perform a **verification step** to review the code and reproduce the results reported by the participants.

- All programs should be executable via a shell script named `mipcomp.sh` (provided by the participants) which receives an MPS file as argument. The following command will be executed for testing

```
sh mipcomp.sh instancename.mps
```

- The output of each run should be a file named `instancename.sol` with the solution found. This is a text file with a list of variable names and values. Each line of this file should be:

```
<variable name> white space <solution value>
```

- The shell script will be executed with a **timeout of 10 minutes** per instance. The code should produce a SOL file during this time.

If significant differences between the submission's report and the jury's verification are detected, the submission will be disqualified and another will take its place.

The submissions that pass the verification step will be run on the **hidden instances**. Based on the written report and these last experiments, a winner will be selected. The jury may also recognize outstanding student submissions.

2 MPC expedited review

High-quality submissions (which may include more than the winner) will be invited to submit full-length manuscripts to Mathematical Programming Computation. These submission will have an **expedited review process**.

3 Evaluation criteria

The evaluation will be performed by a jury consisting of researchers with experience in computational optimization. At both stages, each submission will be judged on the following points:

- **Scope:** On how many instances the submission was able to find a solution.
- **Quality:** On how many instances, compared to other submissions, the best quality solution was found.
- **Runtime:** How time-efficient the algorithm is, regardless of the quality of the solution found.
- **Novelty:** How novel the implemented approach is. In this point, LP-free methods will be positively evaluated.

The spirit of this competition is to encourage the development of new methods that can provide good solutions in practice. The jury will be free to disqualify submissions that grossly violate this target: for instance, hard-coding solutions for known instances will not be accepted.

Questions? Send us an e-mail to mip2022.comp@gmail.com

Competition Organizers
Timo Berthold
Yuri Faenza
Andrés Gómez
Gonzalo Muñoz (chair)