Experimental Reports for *Sparkle*

Sparkle

11th December 2017

1 Introduction

Sparkle [2] is a Programming by Optimisation (PbO) [1] multi-agent problem-solving platform, and would provide many algorithm optimisation techniques (such as automated algorithm configuration, portfolio-based algorithm selection, etc) to light your solvers.

This is the automatically generated report by *Sparkle*, and this report is to present experimental results for the current portfolio selector in *Sparkle* on solving the test instance class **PTN-More**.

2 Information about the Test Instance Class

The test instance class is **PTN-More**, and there are 556 instances in this test instance class.

3 Information about Current *Sparkle*

In this section, we present the experimental preliminaries, including the list of solvers, the list of feature extractors, the list of instance classes, information about experimental setup and the information about how to construct a portfolio-based algorithm selector in *Sparkle*.

3.1 Solvers

There are 7 solver(s) submitted in *Sparkle*, and the list of solvers are given as follows.

1. **PbO-CCSAT-CBMC**
2. **PbO-CCSAT-FAC**
3. **PbO-CCSAT-FCC**
4. **PbO-CCSAT-modgen**
5. **PbO-CCSAT-mp1-9**
6. **PbO-CCSAT-PTN**
7. **PbO-CCSAT-SMT-QF-BV**

3.2 Feature Extractors

There are 1 feature extractor(s) submitted in *Sparkle*, and list of feature extractors are given as follow.

1. **SAT-features-competition2012_sparkle**

3.3 Train Instance Classes

There are 1 instance class(es) submitted in *Sparkle*, and the list of instance classes are given as follows.

1. **PTN-7824_Train**, number of instances: 11

3.4 Experimental Setup for Training Phase

The experimental setup for training phase is described as follows.

Feature computation: *Sparkle* uses all the feature extractors which are presented above to compute the feature vector for each train instance. Each feature extractor computes a feature vector for each train instance. The final feature vector is the combination of all computed feature vectors. The cutoff time for feature vector computation on each train instance is set to 90 seconds.

Performance computation: *Sparkle* runs each solver one time on each train instance. The cutoff time for each performance computation run is set to 5000 seconds.

3.5 Constructing Portfolio-Based Algorithm Selector

Sparkle runs all the feature extractors to compute the feature vector for each train instance, and store the resulting the feature data in the system. Also, *Sparkle* runs all the solvers to solve each train instance, and store the resulting the performance data in the system. After the feature-related and the performance-related experiments done, by utilizing the computed feature data and performance data, *Sparkle* uses *AutoFolio* [3] to automatically construct a portfolio-based algorithm selector for *Sparkle*.

4 Experimental Results

In this section, PAR10 for the current portfolio selector in *Sparkle* on solving the test instance class **PTN-More** is reported as follows.

- **Actual Portfolio Selector in *Sparkle***, PAR10: 11.8454496403

References

[1] Holger H. Hoos. Programming by optimization. *Communications of the ACM*, 55(2):70–80, 2012.

[2] Holger H. Hoos. Sparkle: A pbo-based multi-agent problem-solving platform. Technical report, Department of Computer Science, University of British Columbia, 2015.

[3] Marius Thomas Lindauer, Holger H. Hoos, Frank Hutter, and Torsten Schaub. AutoFolio: An automatically configured algorithm selector. *Journal of Artificial Intelligence Research*, 53:745–778, 2015.