Excavating event logs with DiSCover

Eric Verbeek

Eindhoven University of Technology, Groene Loper 5, 5612 AE Eindhoven, The Netherlands

Abstract

This extended abstract introduces the *Xcavate* ProM plug-in that is built on top of the *DiSCover* plug-in and the *Log Skeleton* filter plug-in. The *Xcavate* plug-in allows the user to specify two ranges of noise levels, to discover a Petri net for every possible combination of noise levels in these ranges, and to return the discovered net that scores best on a combined metric. This combined metric includes a fitness metric, a precision metric, and a simplicity metric. As such, the *Xcavate* plug-in can handle event logs at different noise levels and return a best net.

Keywords

ProM, DiSCover plug-in, Event Logs, Log Skeletons

1. DiSCover

The *DiSCover* plug-in [1] was introduced at the ICPM 2022 conference and introduced a new discovery technique using S-components. The current *DiSCover* plug-in[2] has been improved since then[3] in two ways. First, it now uses by default a minimal set of components that cover all activities. Components not contained in this minimal set are simply ignored. This possibly results in less components, and hence in a simpler Petri net. Second, it now filters out any traces that do not agree with the selected thresholds, and then provides every component with the same filtered log. As a result, every component now uses the exact same set of traces, which results in a Petri net that is relaxed sound.

2. Log Skeleton filter

The *DiSCover* plug-in depends on the directly-follows relation to filter the event log. To compensate for this, the new *Xcavate* plug-in also includes the *Log Skeleton* [2] filter plug-in, which does not depend on this relation. As such, the *Log Skeleton* filter plug-in can filter out types of noise that the *DiSCover* filter cannot filter out. As an example, the discovered log skeleton could contain the constraint that every instance of activity A is eventually followed by an instance of activity B while for only 95% of the instances of activity A this holds. This log skeleton can then be used by the filter to filter out all traces for which some instance of A is not followed by an instance of B.

3. Xcavate

The *Xcavate* plug-in takes the following parameters (see Figure 1):

- *Log skeleton*: A collection of thresholds for the *Log Skeleto*n filter.
- *Relative*: A collection of relative thresholds for the *DiSCover* filter.
- *Fitness*: A (relative) weight for the fitness metric.

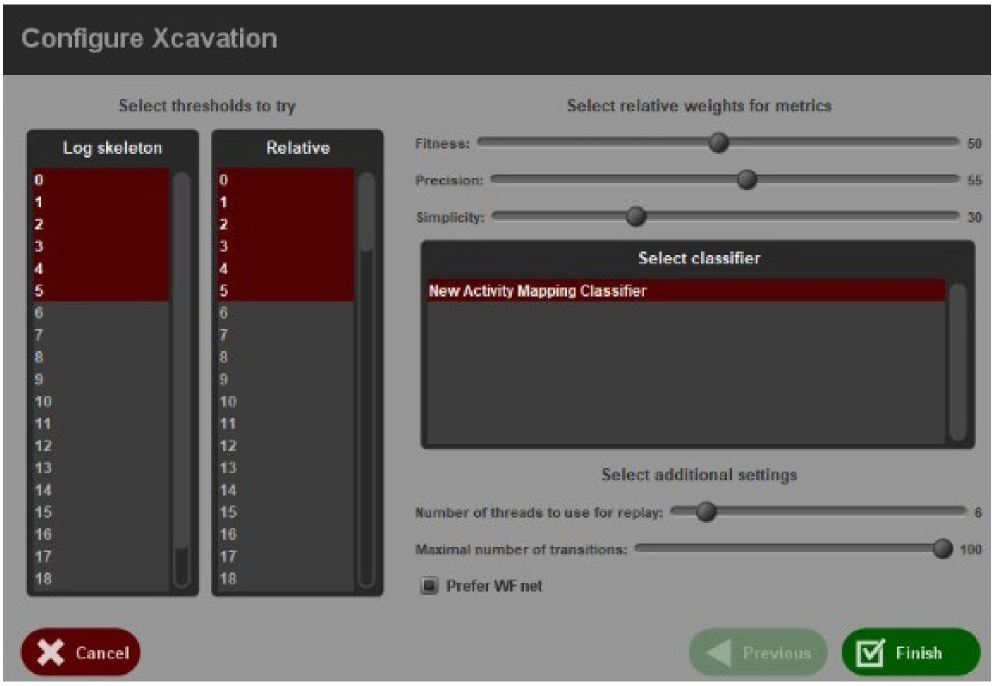
ICPM 2023: Tool demonstrations, October 23–27, 2023, Rome, Italy

h.m.w.verbeek@tue.nl (E. Verbeek)
0000-0002-1658-9679 (E. Verbeek)

CEUR Workshop Proceedings (CEUR-WS.org)

[2] As in the ProM 6.13 release, see https://promtools.org/prom-6-13/.

[3] As in the ProM 6.12 release, see https://promtools.org/prom-6-12/.



Configure Xcavation

Select thresholds to try

Log skeleton	Relative
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16
17	17
18	18

Select relative weights for metrics

Fitness: 50

Precision: 55

Simplicity: 30

Select classifier

New Activity Mapping Classifier

Select additional settings

Number of threads to use for replay: 6

Maximal number of transitions: 100

□ Prefer WF net

Figure 1: The dialog to configure the parameters for the *Xcavate* plug-in.

- *Precision*: A (relative) weight for the precision metric[4].
- *Simplicity*: A (relative) weight for the simplicity metric.
- *Select classifier*: The classifier to use.
- *Number of threads to use for replay*: The number of threads that can be used by the replayer (required to determine fitness and/or precision).
- *Maximal number of transitions*: The maximal number of transitions a Petri net may contain to do a replay.
- *Prefer WF net*: Whether WF nets are preferred over non-WF nets.

The first two parameters determine the ‘excavation site'. For every possible combination of these two thresholds, the plug-in will filter the event log using the *Log Skeleton* filter plug-in and then start the *DisCover* plug-in on the filtered log using the provided relative thresholds. Provided that the discovered Petri net does not contain too many transitions, it will then be scored using the three weights. In the end, the *Xcavate* plug-in returns a Petri net discovered at the excavation site that scores best. If WF nets are preferred, it will return a WF net that scores best.

4. Examples

As a first example, we run the *Xcavate* plug-in on the *BPIC 2012* event log [3] that only contains the A- and O-events, that is, all W-events have been filtered out. For the parameters, we accept the default values, except for the thresholds for which we both select all possible values. We then start the *Xcavate* plug-in by selecting the *Finish* button. The right-hand side of Figure 2 shows the net that is discovered by the plug-in in the end. When compared to the left-hand side (which shows the net as discovered in [1]), we note that the excavated net is indeed relaxed sound, and that it contains fewer (only one instead of eleven) silent transitions.

As a second example, we ran the *Xcavate* plug-in on the *PDC 2022* [4] data set, using the interval [0,5] for the threshold of the *Log Skeleton* filter plug-in and the interval [0,5] for the relative threshold of the *DiSCover* plug-in, and using several different weight configurations (f, p, s)

[4] This metric relies on an optimal alignment, and its actual value may vary depending on the actual optimal alignment. As a result, when using this metric (weight > 0) the excavation may be non-deterministic.

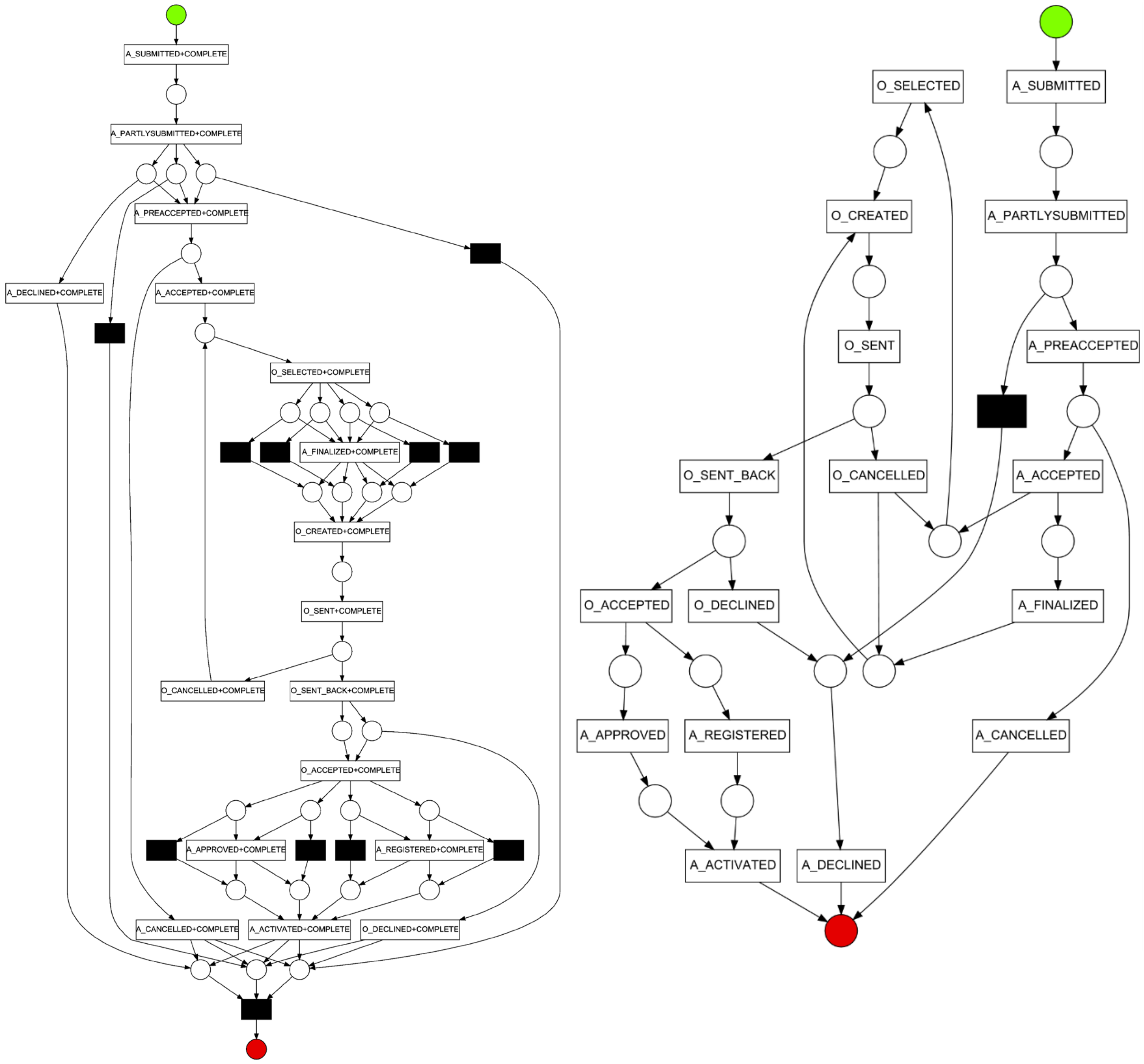


Figure 2: The nets discovered by the original *DiSCover* plug-in (left) and the *Xcavate* plug-in (right) for the BPIC 2012 event log containing only the *A*- and *O*-events.

(where *f* is the fitness weight, *p* the precision weight, and *s* the simplicity weight). For the weight configuration (50,55,30) this resulted in a *PDC 2022* score of 90.67%, which is comparable to the 89.53% of the winner of that contest, and comparable to the 90.41% achieved by the best configuration of the (non-competing) *DiSCover* plug-in back then. As the nets discovered by the new *DiSCover* plug-in are typically much simpler, we can conclude that we get comparable results with simpler nets. As an example, the bottom part of Figure 3 shows the net discovered by the *Xcavate* plug-in for the most complex scenario of the *PDC 2022* data set. When compared to the top part (which shows the net as discovered in [1]), we note that the excavated net is indeed simpler, although it still is too complex to understand from a first glance.

5. Links

From https://www.promtools.org/ you can download the latest releases for ProM. The latest release is ProM 6.13. To use the *Xcavate* plug-in, you need to have the *DiSCover* package (version 6.13.105 or better) installed in ProM.

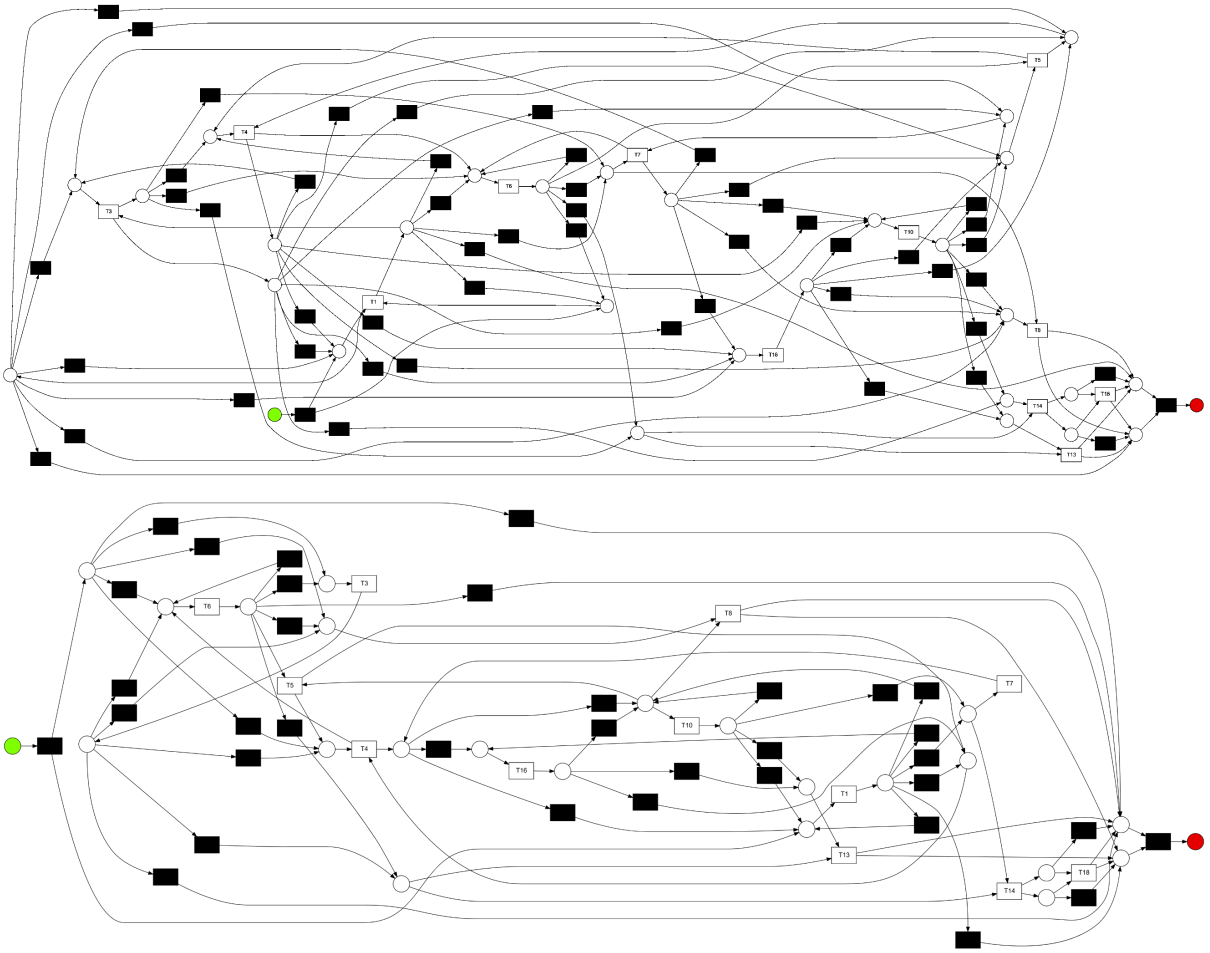


Figure 3: The nets discovered by the original *DiSCover* plug-in (top) and the *Xcavate* plug-in (bottom) for the most complex scenario of the PDC 2022 data set.

You can download a screencast on how to excavate Petri nets from event logs from the *Latest downloads* section on https://hverbeek.win.tue.nl/. In this screencast, we load two event logs, start the *Xcavate* plug-in, configure it, run it, and show the best nets discovered.

On https://svn.win.tue.nl/repos/prom/Packages/DiSCover/Trunk you will find the sources for the *Xcavate* and *DiSCover* plug-ins. ProM and the plug-ins are Open Source.

6. Conclusion

The *Discover* plug-in as introduced during the ICPM 2022 conference has been improved in two ways. First, the discovered nets are now relaxed sound. This is the result of first filtering the log and second constructing the components from the corresponding directly-follows matrix. As a result, all components are discovered from the same set of traces, which results in a relaxed sound net. Second, the discovered nets are now possibly simpler. This is the result of selecting a minimal set of components that covers all activities. Potentially, this could lead to a loss of precision, as a relation between two activities may now be lost as they may not end up in a single component. However, given the fact that the discovered models were often too complex, and supported by the fact that the score for the PDC 2022 contest is still comparable, we feel that this possible loss of precision is acceptable.

The *Xcavate* plug-in has been built on top of the improved *DiSCover* plug-in and extends it with two options. First, it puts the *Log Skeleton* filter plug-in in front of the *DiSCover* plug-in, which allows you to filter out a different of noise than the *DiSCover* plug-in can filter out. Second, it allows the user to provide two collections of noise thresholds: One for the *Log Skeleton* filter plug-in and one for the built-in (relative) filter in the *DiSCover* plug-in. The *Xcavate* plug-in then discovers a net for every possible pair of thresholds and compares the discovered net on its score. This score is the weighted sum of (1) a fitness metric, (2) a precision metric and (3) a simplicity metric. For the fitness metric and the precision metric, the input log will be replayed on the discovered model. The discovered net that results in the best score is then returned by the *Xcavate* plug-in. By setting different weights for the metrics, the user can influence the result.

For future work, we aim to speed up (where possible) the construction of the components. For this construction a step is now required to compute the set of maximal sets of activities that do not directly follow each other. In a later step, this set is reduced to a minimal set of sets that cover all activities. The first step has a bad complexity and may lead to problems. Although for many event logs the running times for this step are perfectly acceptable, for some other event logs this step takes ages. Possibly, by using the fact that we only need a minimal set later, we could simplify the first step and hence alleviate this problem.

References

[1] H. M. W. Verbeek, Discovering an S-Coverable WF-net using DiSCover, in Proceedings of the 2022 4th International Conference on Process Mining (ICPM 2022), A. Burattin, A. Polyvyannyy, and B. Weber, Eds. IEEE, 2022, pp. 64–71. [Online]. Available: https://hverbeek.win.tue.nl/wp-content/papercite-data/pdf/verbeek22.pdf

[2] H. M. W. Verbeek, The Log Skeleton Visualizer in ProM 6.9: The winning contribution to the Process Discovery Contest 2019, International Journal on Software Tools for Technology Transfer, vol. 24, no. 4, pp. 549–561, 2022.

[3] B. van Dongen. (2012, 4) BPI Challenge 2012. https://dx.doi.org/10.4121/uuid:3926db30-f712-4394-aebc-75976070e91f. [Online]. Available: https://data.4tu.nl/articles/dataset/BPI Challenge 2012/12689204

[4] H. M. W. Verbeek, Process Discovery Contest 2022, 4TU.ResearchData, 2022. [Online]. Available: https://icpmconference.org/2022/process-discovery-contest/