On the Flexibility of Regular Process Behaviors
(Extended Abstract)

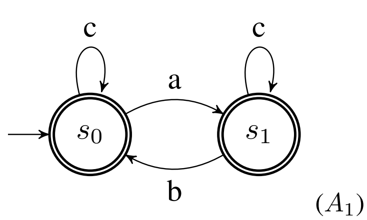
Carl Corea,[1] **Paolo Felli,** [2] **Marco Montali,** [3] **Fabio Patrizi** [4]
[1] Institute for Information Research, University of Koblenz, Germany
[2] Department of Computer Science and Engineering, University of Bologna, Italy
[3] Research Centre for Knowledge and Data, Free University of Bozen-Bolzano, Italy
[4] Department of Computer, Control, and Management Engineering "Antonio Ruberti", Sapienza University Rome, Italy
ccorea@uni-koblenz.de, paolo.felli@unibo.it,
marco.montali@unibz.it, patrizi@diag.uniroma1.it

Abstract

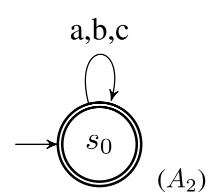
The behaviors allowed by declarative process specifications may be more or less *flexible*, depending on how much freedom they provide. For example, while a specification that allows (only) for repetitions of the activity a may accept an infinite amount of traces ($a, aa, aaaa, ...$), the overall set of behaviors does not provide flexibility, as every action that can be selected at any point in time is predetermined. Surprisingly, little attention has been given to measuring such key features of declarative specifications. In this paper, we close this gap by showing how algorithmic techniques introduced to measure the entropy and the distance of regular languages can be suitably employed to measure the flexibility of infinite regular behaviors. We provide an analysis of the properties of our framework, critically assess its limitations, and show how it can be used to reason about relationships (e.g., inheritance) between different (temporal) declarative specifications.

Keywords. Flexibility, Topological Entropy, Declarative Process Specifications.

The formal modelling of declarative process specification aims at specifying the possible allowed *behaviors* that a process or a system may exhibit (De Giacomo and Rubin 2018; Di Ciccio and Montali 2022). In this regard, the (regular) behaviors captured by the specification of interest may be more or less *flexible*, depending on how much freedom they provide. As an example, consider the two declarative process specifications described by the automata shown in Figure 1.



acb, ccca, cccabccc, ...



aaaa, bbbc, caabbc, ...

Figure 1: Two exemplary declarative process specifications A_1 and A_2, where—as per our definition—A_2 allows for a higher degree of flexibility than A_1.

We see how both automata A_1 and A_2 are constructed over the same alphabet, yet, the automaton A_2 imposes no restriction about the order of elements, which is not the case in A_1. In this sense, the process specification given by A_2 allows for a higher degree of flexibility. Intuitively, with *flexibility* we intend to describe the degree of freedom of choices that can be taken when executing the process. For example, the extreme case of a behavior that does not allow any choice, but simply one predetermined sequence of actions, is considered as non-flexible.

As one can see from the example, the notion of flexibility can provide valuable insights to support the analysis of declarative process specifications, e.g., *"how 'strict' is the specification?"*. However, measuring flexibility has been given little to no attention so far. We thus aim at investigating how to characterise flexibility, providing means for quantitatively assessing the degree of freedom in declarative process specifications.

Specifically, we propose a novel class of *flexibility measures*, which are meant to quantify the degree of freedom provided by a declarative specification with a numerical value – where a higher value indicates a higher level of flexibility.

We start by observing that, for every regular language $\mathcal{L}$ over alphabet Σ, the *maximally permissive behavior* (enjoying the highest degree of freedom) is Σ^*. To clarify, the Σ^*-language allows for any word over Σ without any restriction on how activities should be sequenced therein. We therefore say that Σ^* is the *most* flexible w.r.t. the alphabet Σ. As a second step, we propose to view flexibility as a ratio computed over the maximally permissive behavior, so that a higher value for flexibility conceptually means that a given regular specification is closer to the upper bound given by the Σ^*-language. Using this intuition, we define a baseline flexibility measure as the fraction of words of length n which are supported by a regular language divided by the number of words of length n in Σ^* as n approaches infinity. Here, considering the notation $W_n(\mathcal{L}) = |\{w \in \mathcal{L} \mid \mathit{length}(w) = n\}|$ and $W_{\leq n}(\mathcal{L}) = \sum_{i=0}^{n} W_i(\mathcal{L})$, we define the baseline flexibility measure as follows.

Definition 1. *The flexibility $\mathit{flex}(\mathcal{L})$ of a regular language $\mathcal{L}$ over alphabet Σ is defined via*

$$\mathit{flex}(\mathcal{L}) = \lim_{n \to \infty} \frac{W_{\leq n}(\mathcal{L})}{W_{\leq n}(\Sigma^*)}$$

This proposed approach can provide valuable insights into the flexibility of declarative process specifications. A question remaining in this context is however how to concretely compute this value. In this project, we investigate this problem and show how the $\mathit{flex}()$ value can be computed through a connection with the literature on topological entropy in regular languages (Ceccherini-Silberstein, Machi, and Scarabotti 2003; Parker, Yancey, and Yancey 2017). The concept of topological entropy stems from the research field of symbolic dynamics (Lind and Marcus 2021) and aims to measure a "complexity", or, "capacity" of a language. In essence, topological entropy can be used to determine the growth of distinguishable words of the language up to an arbitrary scale (Polyvyanyy et al. 2020). In this project, we build on this notion and show how it can be exploited to compute the $\mathit{flex}()$ value.

The notion of flexibility envisaged in this project will provide important insights into the "freedom" of the behavior as constrained by declarative process specifications. This will provide valuable additions to declarative process model understanding (Nagel and Delfmann 2022) and declarative process modelling in general. Furthermore, the notion of flexibility will allow for powerful comparison of different declarative process specifications. For example, in the scope of re-modelling declarative process specifications, it can be verified whether any changes to the specification significantly decrease flexibility of the process. Obtaining such an insight would be useful for considering a trade-off between different change operations, and understanding which (behavioral) effect the applied change operations induces over the modified specification. To the best of out knowledge, the only approach that investigates a similar notion of topological entropy (in the context of conformance checking) is by (Polyvyanyy et al. 2020), which however requires to transform the finite behaviour into an infinite one, hence de facto altering the behaviors in the log. We intend to overcome these limitations by lifting the approach presented here to deal with general distance notions starting from the infinite Jaccard distance in (Parker, Yancey, and Yancey 2017).

References

Ceccherini-Silberstein, T.; Machi, A.; and Scarabotti, F. 2003. On the entropy of regular languages. *Theoretical computer science*, 307(1): 93–102.

De Giacomo, G.; and Rubin, S. 2018. Automata-Theoretic Foundations of FOND Planning for LTLf and LDLf Goals. In *Proceedings of 23rd International Joint Conference on Artificial Intelligence (IJCAI'23), Stockholm, Sweden*, 4729–4735.

Di Ciccio, C.; and Montali, M. 2022. Declarative process specifications: reasoning, discovery, monitoring. *Process Mining Handbook. LNBIP*, 448: 108–152.

Lind, D.; and Marcus, B. 2021. *An introduction to symbolic dynamics and coding*. Cambridge university press.

Nagel, S.; and Delfmann, P. 2022. Investigating Inconsistency Understanding to Support Interactive Inconsistency Resolution in Declarative Process Models. In *Proceedings of the 30th European Conference on Information Systems (ECIS'22), Timisoara, Romania*, 1–11.

Parker, A. J.; Yancey, K. B.; and Yancey, M. P. 2017. Regular Language Distance and Entropy. In *Proceedings of the 42nd International Symposium on Mathematical Foundations of Computer Science, Aalborg, Denmark*, 1–14.

Polyvyanyy, A.; Solti, A.; Weidlich, M.; Ciccio, C. D.; and Mendling, J. 2020. Monotone precision and recall measures for comparing executions and specifications of dynamic systems. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 29(3): 1–41.