Behavioral Similarity – A Proper Metric

Matthias Kunze, Matthias Weidlich, Mathias Weske

{matthias.kunze,matthias.weidlich,mathias.weske}@hpi.uni-potsdam.de
Hasso Plattner Institute at the University of Potsdam
Prof.-Dr.-Helmert-Strasse 2-3, 14482 Potsdam

Abstract. With the increasing influence of Business Process Management, large process model repositories emerged in enterprises and public administrations. Their effective utilization requires meaningful and efficient capabilities to search for models that go beyond text based search or folder navigation, e.g., by similarity. Existing measures for process model similarity are often not applicable for efficient similarity search, as they lack metric features. In this paper, we introduce a proper metric to quantify process similarity based on behavioral profiles. It is grounded in the Jaccard coefficient and leverages behavioral relations between pairs of process model activities. The metric is successfully evaluated towards its approximation of human similarity assessment.

1   Introduction

Business Process Management (BPM) found its way into enterprises and public administrations likewise, which led to the advent of large process model repositories. Organizations, both public and private, maintain collections of thousands of process models [1,2]: The Dutch government maintains a set of about 600 reference processes[1] and the German government strives to establish a platform to access and maintain public administration process models[2].

The effective use of large model repositories requires means to access and manage models, and fast search methods in particular. Process modelers want to search a repository for similar processes to avoid the creation of duplicates. Also, reuse of existing knowledge, avoiding redundant work, or establishing reference models requires powerful means to search existing information. In practice, these needs are addressed only by simplistic search features, e.g., folder based navigation or text based search. Exact match search of process models is often not desired, due to the high heterogeneity of modeling languages, guidelines, and terminology. Similarity search also aims at addressing this problem.

In general, the search problem is constrained by three factors: (1) the type of data that is searched for, (2) the method of comparing individual instances of this data, and (3) the specification of the search query [3]. In process model similarity search, (1) and (3) are expressed alike, i.e., one searches for process models that

---

[1] Documentair StruktuurPlan: `http://www.model-dsp.nl/`
[2] Nationale Prozessbibliothek: `http://www.prozessbibliothek.de/`

are similar to a given query model. Considering (2), there have been various proposals to assess process model similarity: based on textual information, the structure of process models, and their execution semantics. Still, most of these approaches rely on exhaustive searching, i.e., the query model is compared with each model in the repository. Well-established techniques for indexing cannot be applied in most cases, as the measures lack metric properties. That is, they are missing a distance function that features the triangle inequality and provides transitivity semantics for pairwise distances of process models.

In this paper, we propose a behavioral metric that quantifies the similarity of process models. It enables efficient similarity search as it satisfies metric properties, in particular the triangle inequality [3]. The metric builds on behavioral profiles, an abstraction of the behavior of a process model [4]. These profiles capture constraints on the execution order of pairs of activities, such as exclusiveness or strict order. Using this abstraction, we propose five elementary similarity measures. Based on these measures, we construct a metric that quantifies the behavioral similarity of process models. As an evaluation, we conducted experiments using the SAP reference model and manual similarity judgments by BPM experts. Our metric shows a good approximation of human similarity assessment. We sketched the idea of leveraging behavioral profiles for similarity search in [5]. In this paper, we formally introduce a more advanced metric and evaluate it experimentally.

The remainder of this paper is structured as follows: The next section gives on overview of similarity search and process model similarity. We introduce the formal preliminaries in Section 3. Our metric to quantify process model similarity is introduced in Section 4. We present the results of an experimental evaluation in Section 5. Related work is discussed in Section 6, before we conclude the paper and give an outlook on future work in Section 7.

2 Background

Search involves comparison of a query with stored data objects to decide, whether these objects shall be included in the search result. Its complexity can be estimated by the number of required comparison operations multiplied by the complexity of the comparison operation. Similarity search algorithms make search more efficient by significantly reducing the number of comparison operations. Thus, we review principles thereof in Section 2.1, before we explore existing measures for process model similarity and determine whether they can be applied to the aforementioned principles in Section 2.2.

2.1 Similarity Search

Traditional databases have been tailored to execute searches on structured data efficiently. They use tree or hash based indexes to quickly access data. Contemporary data, e.g., process models, cannot be mapped to such search structures in a meaningful way, because there exists no natural ordering among the data objects and hashing does not expose any expressive classification. Instead,
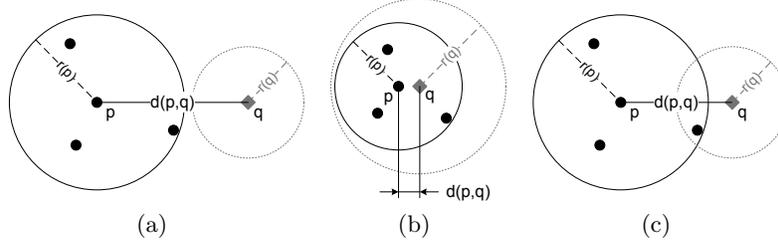
Fig. 1. Metric space partition $T(p)$ (solid circle) with pivot p and similarity query (dotted circle) with query model q. (a) Exclusion, (b) Inclusion, (c) Intersection.

data transformation and reduction techniques need to be applied, which often result in objects for which only pairwise similarity can be computed [3].

Within the last two decades, data structures and algorithms have been studied that allow for efficient similarity search in such a case, cf. [6,7]. These methods leverage transitivity properties exposed by a distance function—a complement to the similarity of data objects—a *metric*.

Definition 1 (Metric). *A metric is a distance function $d : \mathcal{D} \times \mathcal{D} \to \mathbb{R}$ between objects of domain $\mathcal{D}$ with the following properties:*
  - *Symmetry: $\forall o_i, o_j \in \mathcal{D} : d(o_i, o_j) = d(o_j, o_i)$*
  - *Nonnegativity: $\forall o_i, o_j \in \mathcal{D}, o_i \neq o_j : d(o_i, o_j) > 0$*
  - *Identity: $\forall o_i, o_j \in \mathcal{D} : d(o_i, o_j) = 0 \Leftrightarrow o_i = o_j$*
  - *Triangle inequality: $\forall o_i, o_j, o_k \in \mathcal{D} : d(o_i, o_k) \leq d(o_i, o_j) + d(o_j, o_k)$*

A metric space is a pair $\mathcal{S} = (\mathcal{D}, d)$.

The *triangle inequality* enables determining minimum and maximum distances of two objects o_i, o_k without calculating it, if their pairwise distance to a third object o_j is given. To search efficiently in large data sets, i.e., avoid comparison with every object, the data set is split into partitions, from which some can be pruned during search. In metric spaces, a partition is established by a pivot $p \in \mathcal{D}$ and a *covering radius* $r(p) \in \mathbb{R}$ that spans a sphere around p. All data objects o_i within a distance $d(o_i, p) \leq r(p)$ are contained in that sphere, $o_i \in T(p)$.

Similarity search requires a query q from the same domain of the objects in the data set, i.e., $q \in \mathcal{D}$, and a tolerance that expresses how similar the objects of a valid result may be to q—a *query radius* $r(q) \in \mathbb{R}$. Fig. 1(a) shows a pivot p with its covering radius $r(p)$ and all elements of $T(p)$ (solid circle), along with the query model q and its similarity tolerance (dotted circle) spanned by $r(q)$.

Search in metric spaces is efficient, as some partitions are excluded without comparing any contained element with the given query, i.e., the number of comparison operations is reduced compared to exhaustive search. For each partition, the distance between the pivot and a query model $d(p, q)$ is calculated by the distance metric. Because the distances between q and each object in $T(p)$, including p, obey the triangle inequality, one can prune partitions from further examination:

$r(q) + r(p) < d(p, q)$: All objects in $T(p)$ are further away from q than $r(q)$, i.e., they cannot be in the result set, cf. Fig. 1(a). The distance from q to any other element in $T(p)$ does not have to be calculated.

$r(q) - r(p) \leq d(p, q)$: $T(p)$ is completely included in the sphere around q, cf. Fig. 1(b). Thus, all elements satisfy the similarity constraint and need not be compared with the query unless a ranking of the search result is desired.

$r(q) + r(p) \geq d(p, q)$: The spheres of p and q intersect, cf. Fig. 1(c). Identification of the objects that lie in this intersection requires exhaustive search of $T(p)$.

Pruning complete partitions from search reduces the time complexity of search, i.e., the number of comparison operations. Indexing techniques use these capabilities to implement efficient search [6,7]. Here, the limiting factor is the complexity of the comparison operation, i.e., the distance metric.

2.2 Process Model Similarity

Similarity of process models is evaluated based on three complementary aspects, i.e., the element labels, the graph structure, or the execution semantics [8].

To determine the similarity of labels, techniques from schema matching [9] and ontology matching [10] are used. Labels are compared on the syntactic or semantic level. The string edit distance (SED) [11] is a well-known example for the former. It counts the minimal number of atomic character operations (insert, delete, update) needed to transform one string into another. For instance, the SED for the labels 'file order' and 'filed the order' is five. Other measures do not compare strings as a whole, but tokenize strings into a bag of terms. Then, distances in a vector space are used to judge the similarity of two strings [12]. Various approaches to label matching rely on Natural Language Processing techniques [13] to compensate for heterogeneous terminology, e.g., term stemming, stop-word elimination, or external knowledge such as WordNet [14]. Difference in the granularity of two models can be handled by automatic approaches only to very limited degree. Therefore we resort to comparing only single node's labels. Most of the existing approaches to structural or behavioral process model similarity incorporate some kind of label matching to judge the similarity of model elements.

Approaches to structural similarity of process models leverage the maximum common sub-graph isomorphism and the graph edit distance (GED). The latter defines the minimal number of atomic graph operations (substitute node, insert/delete node, (un)grouping nodes, substitute edge, insert/delete edge) needed to transform one graph into another [15]. The GED problem is NP-hard [16], so that search algorithms and heuristics are applied to compute the distance. The GED has been used to score the similarity of process models in [17,2]. In the same vein, structural differences between process models may be grouped to change operations to determine the similarity of two process models [18]. Further, classification of process model elements according to the cardinality of their incoming and outgoing flows has been used for similarity assessment [19].

Similarity measures have also been defined on the sets of all traces of process models. The size of the intersection of these sets relative to the overall number of traces would be a straight-forward example for such a behavioral measure [8].

Table 1. Overview of process model similarities

Approach	Aspect	Symmetry	Nonnegativity	Identity	Triangle inequality
Minor et al. [24]	Structure	yes	yes	yes	no
Li et al. [18]	Structure	yes	yes	yes	no
Ehrig et al. [25]	Structure	yes	yes	yes	no
Dijkman et al. [17]	Structure	yes	yes	yes	yes
Eshuis and Grefen [22]	Behavior	yes	yes	yes	yes
Aalst et al. [26]	Behavior	no	yes	yes	no
Wombacher and Rozie [20]	Behavior	yes	yes	yes	no
Lu and Sadiq [27]	Structure	yes	yes	yes	yes
Dongen et al. [23]	Behavior	yes	yes	yes	no
Nejati et al. [21]	Behavior	yes	yes	yes	no
Yan et al. [19]	Structure	yes	yes	yes	no

Further, edit distances have been defined for the behavior of a process model based on all traces, an automaton encoding the language, or an n-gram representation of all traces [20]. If the process behavior is defined by a transition system, the degree to which two systems simulate each other can be used as a similarity measure [21]. Other measures exploit behavioral abstractions. Behavioral relations defined over pairs of activities, e.g., order and exclusiveness, provide a means to enrich structural matching [22]. Causal footprints have been proposed to approximate the behavior of process models to assess their similarity [23]. Those footprints capture causal dependencies for activities by sets of causal predecessors and successors for each activity.

For each of the discussed similarities, Table 1 lists whether the measures meet the metric properties, cf. Definition 1. For none of the approaches, the authors actually proved metric properties, so that the Table 1 shows our informal evaluation of the proposed measures. Most measures turn out to be semi-metrics, i.e., violate the triangle inequality. Hence, they cannot be applied to search a metric space efficiently. There are few notable exceptions. The measure based on the graph edit distance [17] features the metric properties, see also [28]. Still, this measure is restricted to the model structure. The pairwise set similarity of process model features (e.g., nodes and edges) applied in [27] yields a metric, but neglects the graph structure completely. Although not proven formally, the measure proposed for BPEL processes [22] seems to satisfy the triangle inequality. We later discuss that our metric can be seen as a generalization of this approach.

We conclude that there are virtually no measures available that are based on the behavior of a process model and satisfy all metric properties.

3   Preliminaries

For our work, we rely on a notion of a process model that comprises activity nodes and control nodes. It captures the commonalities of many process modeling languages, such as BPMN.

Definition 2 (Process Model).
A process model *is a tuple $P = (A, s, e, C, N, F, T)$ where:*
- *A is a finite non-empty set of activity nodes, C is a finite set of control nodes, and $N = A \cup C$ is a set of nodes with $A \cap C = \emptyset$,*
- *$F \subseteq N \times N$ is the flow relation,*
- *$\bullet n = \{n' \in N | (n', n) \in F\}$ and $n \bullet = \{n' \in N | (n, n') \in F\}$ denote direct predecessors and successors, we require $\forall\, a \in A :\, | \bullet a| \leq 1 \wedge |a \bullet | \leq 1$,*
- *$s \in A$ is the only start node, $\bullet s = \emptyset$, and $e \in A$ is the only end node, $e \bullet = \emptyset$,*
- *$(N, F \cup \{(e, s)\})$ is a strongly connected graph,*
- *$T : C \to \{and, xor\}$ associates each control node with a type.*

Our notion requires dedicated start (s) and end (e) activities. Refactoring techniques may be applied to normalize models that do not match this assumption [29]. For a process model, we assume trace semantics. The behavior of a process model $P = (A, s, e, C, N, F, T)$ is a set of traces $\mathcal{T}_P$. A trace is a list $\sigma = \langle s, a_1, a_2, \ldots \rangle$, $a_i \in A$ for all $0 < i$. It represents the order of execution of activities, as it follows on common Petri net-based formalizations [30].

Our metric relies on behavioral profiles as a behavioral abstraction. These profiles capture behavioral characteristics of a process model by means of relations between activity pairs. These relations are grounded in weak order, which holds between two activities if both are observed in a trace in a certain order.

Definition 3 (Weak Order). *Let $P = (A, s, e, C, N, F, T)$ be a process model and $\mathcal{T}_P$ its set of traces. The* weak order *relation $\succ_P \subseteq (A \times A)$ contains all pairs (x, y), such that there exists a trace $\sigma = \langle a_1, a_2, \ldots \rangle$ in $\mathcal{T}_P$ and there exist two indices $j, k \in \{1, 2, \ldots\}$ with $j < k$ for which holds $a_j = x$ and $a_k = y$.*

Using the notion of weak order, we define the behavioral profile as follows.

Definition 4 (Behavioral Profile). *Let $P = (A, s, e, C, N, F, T)$ be a process model. A pair $(x, y) \in (A \times A)$ is in the following relations:*
- *The* strict order *relation $\leadsto_P$, iff $x \succ_P y$ and $y \not\succ_P x$.*
- *The* exclusiveness *relation $+_P$, iff $x \not\succ_P y$ and $y \not\succ_P x$.*
- *The* interleaving order *relation $||_P$, iff $x \succ_P y$ and $y \succ_P x$.*
$\mathcal{B}_P = \{\leadsto_P, +_P, ||_P\}$ is the behavioral profile *of P.*

For each pair (x, y) in strict order, the *reverse strict order* relation comprises the inverse pair (y, x), i.e., $x \leadsto_P y \Leftrightarrow y \leadsto_P^{-1} x$. Behavioral profiles show a number of properties that we will exploit for similarity analysis. Together with the reverse strict order relation, the relations of the behavioral profile partition the Cartesian product of activities [4]. Further, behavioral profiles are computed efficiently for the class of process models introduced above under the assumption of soundness. Soundness is a correctness criterion that guarantees the absence of behavioral anomalies, see [31]. Our notion of a process model translates into a free-choice WF-net [30], a dedicated structural sub-class of Petri nets, which may involve adding fresh transitions to the WF-net. Thus, we can apply the soundness criterion to process models directly. We also reuse the computation techniques for behavioral profiles introduced for sound free-choice WF-nets. This

allows for the computation of behavioral profiles for sound process models in cubic time to the size of the model [4].

4 A Spectrum of Behavioral Similarity Metrics

This section introduces how behavioral profiles are used to measure similarity between a pair of process models. We propose a set of five elementary similarity measures and explain how we construct a metric from these measures.

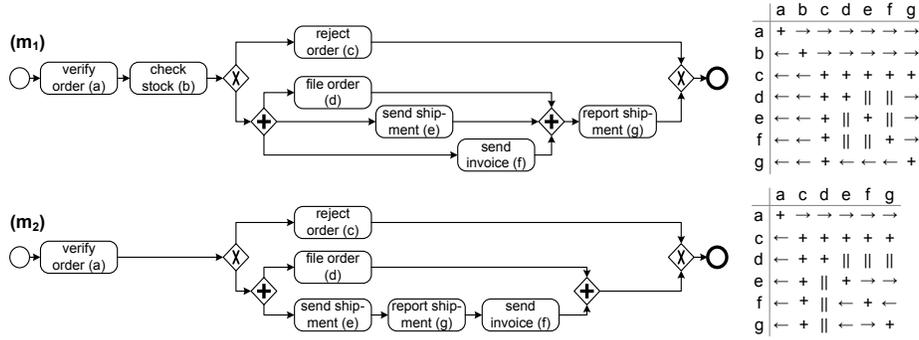### 4.1 Similarity based on Behavioral Profiles

Behavioral profiles as introduced in the previous section provide a behavioral abstraction. The notion focuses on the *order of potential execution* of activities and neglects other behavioral details. Causality between activities and cardinality constraints on their execution are abstracted, cf. [32]. Also, the interleaving order relation does not differentiate whether activities share a loop or are on concurrent paths. Findings from experiments with process variants [4] and empirical work on the perception of consistency between process models by process analysts [33] suggest that behavioral profiles capture a significant share of the behavioral characteristics of a process model. This provides us with evidence that implied information loss is suited to be traded for computational efficiency of the search technique. We later challenge this assumption using an experimental setup.

Similarity assessment based on behavioral profiles requires matching activities. Given two process models, correspondences between activities have to be determined. These correspondences are used to quantify the overlap of behavior in both models. In Section 2.2, we provided an overview of techniques to identify corresponding pairs of activities. Our work does not focus on this aspect of similarity, but rather on behavioral properties. Therefore, we assume these correspondences to be given, hereafter. To keep the formalization of our metrics concise, we abstract from such correspondences and assume corresponding activities to be identical. In other words, given two process models P and Q with their sets of activities A_P and A_Q, a correspondence between an activity in P and an activity in Q is manifested as the existence of an activity $a \in (A_P \cap A_Q)$.

4.2 Elementary Similarity Measures

A process model P resembles another model Q in certain behavioral aspects if they overlap in their behavioral profiles $\mathcal{B}_P$ and $\mathcal{B}_Q$, respectively. The larger this overlap is, the more similar we assume these models to be. We quantify similarity by the well-known Jaccard coefficient for two sets: $sim(A, B) = \frac{|A \cap B|}{|A \cup B|}$. First, this measure can easily be applied to behavioral profiles, since each relation of a behavioral profile is essentially a set. Second, it can be translated into a metric $d(A, B) = 1 - sim(A, B)$ [34] to enable efficient similarity search.

Fig. 2 shows two order handling process models (m_1, m_2), in which an order is received, checked, and fulfilled or rejected. The correspondences between activities

	a	b	c	d	e	f	g
a	+	→	→	→	→	→	→
b	←	+	→	→	→	→	→
c	←	←	+	+	+	+	+
d	←	←	+	+	‖	‖	→
e	←	←	+	‖	+	‖	→
f	←	←	+	‖	‖	+	→
g	←	←	+	←	←	←	+

	a	c	d	e	f	g
a	+	→	→	→	→	→
c	←	+	+	+	+	+
d	←	+	‖	‖	‖	‖
e	←	+	‖	+	→	→
f	←	+	‖	←	+	←
g	←	+	‖	←	→	+

Fig. 2. Order management process models m_1 and m_2 with their behavioral profile matrices $\mathcal{B}_{m1}$ and $\mathcal{B}_{m2}$.

are illustrated by equal labels in both models, and are represented through the indices a to g. The behavioral profiles ($\mathcal{B}_{m1}$, $\mathcal{B}_{m2}$) are depicted as matrices over the activities, e.g., a and g are in strict order, since in all traces, where both activities appear, a happens before g.

Exclusiveness is the strictest relation of a behavioral profile, because it enforces the absence of a co-occurrence of two activities within one process instance. The corresponding similarity quantifies, how many common pairs exist in two process models that feature the same exclusiveness relation.

Definition 5 (Exclusiveness Similarity). *Let P, Q be process models and* $+_P$, $+_Q$ *the exclusiveness relations of their respective behavioral profiles* $\mathcal{B}_P$, $\mathcal{B}_Q$. *We define the* Exclusiveness Similarity

$$sim_+(\mathcal{B}_P, \mathcal{B}_Q) = \frac{|+_P \cap +_Q|}{|+_P \cup +_Q|}$$

See, for example, the activities c and d in both process models of Fig. 2. From the scenario, it is prohibitive to reject an order and file this order and fulfill it. The only difference between m_1 and m_2 with regard to the exclusiveness relation is the absence of b in m_2, because this activity is exclusive to itself. This yields a high similarity $sim_+(\mathcal{B}_{m1}, \mathcal{B}_{m2}) = \frac{14}{15} \approx 0.933$.

In most cases, the order of tasks of a business process is reflected by the strict order of these activities in the process model. The strict order similarity strives to reward a large overlap of two strict order relations, whereas order violations are penalized.

Definition 6 (Strict Order Similarity). *Let P, Q be process models and* $\rightsquigarrow_P$, $\rightsquigarrow_Q$ *the strict order relations of their respective behavioral profiles* $\mathcal{B}_P$, $\mathcal{B}_Q$. *We define the* Strict Order Similarity

$$sim_\rightsquigarrow(\mathcal{B}_P, \mathcal{B}_Q) = \frac{|\rightsquigarrow_P \cap \rightsquigarrow_Q|}{|\rightsquigarrow_P \cup \rightsquigarrow_Q|}$$

Due to $x \rightsquigarrow_P y \Leftrightarrow y \rightsquigarrow_P^{-1} x$, cf. Section 3, it suffices to incorporate only the strict order relation into the similarity measure. The reverse relation is implicitly covered. This can be seen in the behavioral profile matrix as the strict order

relations are mirrored along the diagonal axis, e.g., in m_2 the pair (f,g) is in reverse strict order and (g,f) is in strict order.

The exemplary process models m_1 and m_2 show a significant difference in their strict order relations, because the activities e and f are in interleaving order in m_1, whereas in m_2 they are in strict order. Additionally, activity b is missing in m_2, which yields five additional pairs for the strict order relation in $\mathcal{B}_{m1}$. The strict order relations between (a,c), (a,d), (a,e), (a,f), and (a,g) are, however, not affected by that. This leads to the following strict order similarity: $sim_{\rightsquigarrow}(\mathcal{B}_{m1}, \mathcal{B}_{m2}) = \frac{6}{16} = 0.375$.

Interleaving order is the weakest imposition on the relations between two activities, since it only states that they may be executed in any order in one process instance. Thus, the interleaving order similarity also rewards matching pairs, if they are, e.g., executed in parallel in one process and as part of a control flow cycle in the other.

Definition 7 (Interleaving Order Similarity). *Let P, Q be process models and $||_P$, $||_Q$ the interleaving order relations of their respective behavioral profiles $\mathcal{B}_P$, $\mathcal{B}_Q$. We define the* Interleaving Order Similarity
$sim_{||}(\mathcal{B}_P, \mathcal{B}_Q) = \frac{|\,||_P \cap ||_Q\,|}{|\,||_P \cup ||_Q\,|}$

The process models of Fig. 2 share interleaving order relations for (d,e),(e,d), (d,f) and (f,d). This yields $sim_{||}(\mathcal{B}_{m1}, \mathcal{B}_{m2}) = \frac{4}{8} = 0.5$.

Not in all cases does the ordering of activities in a model correspond to the actual order of execution of these activities in practice. Their order may simply be one possible sequence sprung from the habit of the process modeler. To address such cases, we extend the elementary similarities above to increase the tolerance of the measures.

The strict order similarity only rewards pairs of activities that are executed in the same order. If two activities appear both in each trace of two distinct process models but are executed in inverse order respectively, it is plausible that they do not depend on each other, but rather have simply been modeled in a sequence that seemed suitable to the process modeler. Thus, we propose the extended strict order similarity that will reward these pairs of activities.

Definition 8 (Extended Strict Order Similarity). *Let P, Q be process models and $\rightsquigarrow_P$, $\rightsquigarrow_Q$ the strict order relations, $\rightsquigarrow_Q^{-1}$, $\rightsquigarrow_P^{-1}$ the reverse strict order relations of their respective behavioral profiles $\mathcal{B}_P$, $\mathcal{B}_Q$. We define the* Extended Strict Order Similarity
$sim_{\rightsquigarrow'}(\mathcal{B}_P, \mathcal{B}_Q) = \frac{|(\rightsquigarrow_P \cup \rightsquigarrow_P^{-1}) \cap (\rightsquigarrow_Q \cup \rightsquigarrow_Q^{-1})|}{|(\rightsquigarrow_P \cup \rightsquigarrow_P^{-1}) \cup (\rightsquigarrow_Q \cup \rightsquigarrow_Q^{-1})|}$

The above consideration applies to activities f and g in Fig. 2: They are in strict order in m_1 and in reverse strict order in m_2. In the given scenario, it seems to be reasonable to assume that there is no explicit order constraint between them. Thus, these pairs should contribute to the similarity of both models: $sim_{\rightsquigarrow'}(\mathcal{B}_{m1}, \mathcal{B}_{m2}) = \frac{14}{30} \approx 0.467$.

The interleaving order relation will only identify pairs of activities that are neither in strict order nor exclusive to each other. However, the execution of a

pair of activities in interleaving order in one model also resembles the execution of the same activities in a static sequence in another model. That is, interleaving order supersedes strict order in flexibility. However, this is not supported by the interleaving order similarity. Thus, we suggest the extended interleaving order similarity, which rewards the containment of strict order execution in interleaving order, accounting for strict order in both behavioral profiles of models P and Q in both directions, i.e., in forward and reverse strict order.

Definition 9 (Extended Interleaving Order Similarity). *Let P, Q be process models and $\leadsto_P$, $\leadsto_Q$ the strict order relations, $\leadsto_P^{-1}$, $\leadsto_Q^{-1}$ the inverse strict order relations, $||_P$, $||_Q$ the interleaving order relations of their respective behavioral profiles $\mathcal{B}_P$, $\mathcal{B}_Q$. We define the* Extended Interleaving Order Similarity

$$sim_{||'}(\mathcal{B}_P, \mathcal{B}_Q) = \frac{|(\leadsto_P \cup \leadsto_P^{-1} \cup ||_P) \cap (\leadsto_Q \cup \leadsto_Q^{-1} \cup ||_Q)|}{|(\leadsto_P \cup \leadsto_P^{-1} \cup ||_P) \cup (\leadsto_Q \cup \leadsto_Q^{-1} \cup ||_Q)|}$$

As an example, refer to the activities `e` and `f` in Fig. 2. They are in interleaving order in m_1. In an instance of this process they may be executed in the order (`e`,`f`), which resembles the strict order relation of these activities in m_2. Analogously, this holds for activities `d` and `g`, which yields $sim_{||'}(\mathcal{B}_{m1}, \mathcal{B}_{m2}) = \frac{20}{36} \approx 0.556$.

In contrast to pure ordering features, i.e., strict order and interleaving order, we do not relax the exclusiveness similarity, since exclusiveness is a very strong statement about the dependency and correlation of activities.

4.3 Aggregated Metric for Behavioral Profiles

Based on the elementary similarity measures defined above, we construct an aggregated similarity metric as follows. Each elementary similarity translates into an elementary metric, $d_h(\mathcal{B}_P, \mathcal{B}_Q) = 1 - sim_h(\mathcal{B}_P, \mathcal{B}_Q)$ for all $h \in \{+, \leadsto, ||, \leadsto', ||'\}$, as explained in Section 4.2 and [34]. Then, we sum up these values and assign a weight that accounts for the respective metric's impact on the overall metric.

We postulate $\mathcal{B}$ as the universe of all possible behavioral profiles. In practice this matches the behavioral profiles of all models within a repository.

Definition 10 (Behavioral Profile Metric). $\mathfrak{B} = (\mathcal{B}, d_{\mathcal{B}})$ *is a metric space of behavioral profiles $\mathcal{B}$, where the* behavioral profile metric $d_{\mathcal{B}} : \mathcal{B} \times \mathcal{B} \to \mathbb{R}$ *is a metric,*

$$d_{\mathcal{B}}(\mathcal{B}_P, \mathcal{B}_Q) = 1 - \sum_h w_h \cdot sim_h(\mathcal{B}_P, \mathcal{B}_Q)$$

with $h \in \{+, \leadsto, ||, \leadsto', ||'\}$ and weighting factors $w_h \in \mathbb{R}, 0 < w_h < 1$ such that $\sum_h w_h = 1$.

In order to use this aggregate metric for similarity search, it has to be proven that it is indeed a metric as established in Definition 1.

Theorem 1. *The weighted sum $D(o_i, o_j) = \sum w_h \cdot d_h(o_i, o_j)$ of elementary metrics d_h is a metric if $\forall h \in [1...n] : w_h \in \mathbb{R} \wedge 0 < w_h < 1$, $\forall o_i, o_j, o_k \in \mathcal{D}$, and $d_h(o_i, o_j) \in \mathbb{R}$.*

Proof. $D(o_i, o_j)$ holds the properties *symmetry*, *nonnegativity*, *identity*, and *triangle inequality*.

Symmetry: From $d_h(o_i, o_j) = d_h(o_j, o_i)$ it follows directly from the commutativity of the summation operation that $D(o_i, o_j) = D(o_j, o_i)$.

Nonnegativity: From $d_h(o_i, o_j) \geq 0$ and $w_h > 0$ it follows that $w_h \cdot d_h(o_i, o_j) \geq 0$ and thus their sum $D(o_i, o_j) \geq 0$.

Identity: From $d_h(o_i, o_i) = 0$ it follows directly that $D(o_i, o_i) = 0$ and if $D(o_i, o_j) = 0$ all $d_h(o_i, o_j) = 0$ because $w_h \cdot d_h(o_i, o_j) \geq 0$ with $w_h > 0$. From $d_h(o_i, o_j) = 0$ however, it follows that $o_i = o_j$ and thus $D(o_i, o_j) = 0 \Leftrightarrow o_i = o_j$.

Triangle Inequality: From $d_h(o_i, o_k) \leq d_h(o_i, o_j) + d_h(o_j, o_k)$ and $w_h > 0$ it follows $w_h \cdot d_h(o_i, o_k) \leq w_h \cdot d_h(o_i, o_j) + w_h \cdot d_h(o_j, o_k)$ and through summation $D(o_i, o_k) \leq D(o_i, o_j) + D(o_j, o_k)$. $\qquad\square$

5 Experimental Evaluation

This section presents an experimental evaluation of the proposed similarity measures and the aggregated metric. We describe the setup to challenge them in Section 5.1. Section 5.2 presents the results of our evaluation.

5.1 Setup

Our experiment relies on a collection of process models that has been used to test similarity measures in [2]. We took the test set for the 'evaluation with homogeneous labels' [2] as our focus is on behavioral similarity. The collection comprises 100 randomly selected models from the SAP reference model, called *document models*. 10 models have been selected as *query models*. Two of those models were left unchanged, the others underwent slight modifications such as extracting a subgraph or changing types of control flow routing elements, see also [2]. The authors of [2] provide a relevance mapping that assigns to each query model, the document models that were found relevant by humans (this scoring involved process modeling experts). Using this information, similarity measures can be tested for their approximation of the human similarity assessment. Such tests have been conducted for measures based on the graph edit distance and based on causal footprints [2], which we discussed in Section 2.2.

For our experiment, 15 out of 100 models had to be sorted out because of ambiguous instantiation semantics. These models have multiple start events that are not merged into a single join connector, which is also referred to as a *start join* [35]. We manually checked all models without a start join and identified 15 models with ambiguous instantiation semantics. One of these models was a query model, so that our setup comprises 85 document models and nine query models. Behavioral profiles do not discover order constraints in cyclic structures. However, in our collection, we observed only 3 models with control flow cycles.

We evaluated the similarity of model elements based on the string edit distance similarity of their labels. If this similarity exceeds a threshold (we used a threshold
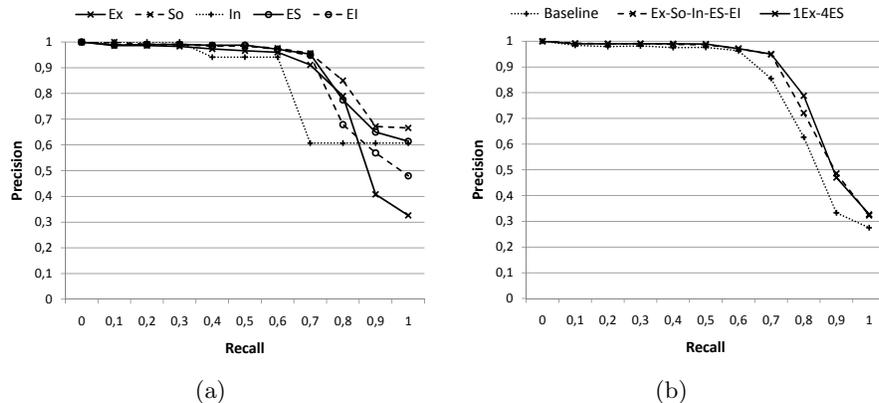
Fig. 3. (a) Precision-recall curve for metrics based on elementary similarities, (b) precision-recall curve for a baseline metric, and two aggregated metrics

of 0.6), two elements are considered to be a matching pair. This proves very effective for the given set of process models, which shows a very homogeneous vocabulary, cf. [2].

5.2 Evaluation

First, we evaluated measures that are grounded in one of the elementary similarities presented in Section 4.2. Hence, only one aspect, e.g., exclusiveness, is considered for a similarity assessment. For all nine query models, we obtained a ranked list of query results. We computed the average precision, i.e., the average of the precision values obtained after each relevant document model is found [36]. Aggregating the results for all queries using the arithmetic mean yields the following mean average precision values.

- Exclusiveness Similarity (Ex): 0.70
- Strict Order Similarity (So): 0.75
- Interleaving Order Similarity (In): 0.29
- Extended Strict Order Similarity (ES): 0.75
- Extended Interleaving Order Similarity (EI): 0.76

We obtained the best results with a similarity assessment based on strict order and exclusiveness. The difference between strict order similarity and interleaving order similarity shows that the good results obtained for extended interleaving order similarity are grounded on the strict order. This is reasonable against the background discussed in Section 4.2. Exclusiveness and strict order can be seen as the strictest relations of the behavioral profile, so that they are most distinguishing.

The mean average precision provides a rather compact view on the quality of the similarity assessment. We also investigated the relation between precision and recall for all metrics. For ten recall values in the interval between zero and

one, Fig. 3(a) depicts the precision values obtained with the different metrics. Even though the strict order similarity yields good overall results, it does not achieve the best precision values for all recall levels. This suggests applying an aggregated similarity that combines several elementary metrics in a weighted fashion as introduced in Section 4.3.

We evaluated aggregated metrics based on behavioral profiles in a second experiment. The result is illustrated in Fig. 3(b) as a precision-recall curve for three metrics. First, as a baseline metric, we used the set similarity applied to the matching activities of two process models (data series Baseline). Second, we applied an aggregated metric that combined all elementary similarities with equal weights (Ex-So-In-ES-EI). Third, based on the results obtained in the previous experiment, we chose the exclusiveness and extended strict order similarities and combined them into an aggregated metric. For the respective weights, we tested a spectrum of weights for the two similarities. The data series 1Ex-4ES in Fig. 3(b) relates to the similarity that assigns four times the weight to extended strict order. Fig. 3(b) illustrates that our metrics perform better than the simple structural assessment using set similarity for matching activities. Still, the increase in precision for high recall levels is modest, which is likely to be caused by the homogeneity of the used model collection. Also, the precision-recall curves suggest that for this model collection, the differences between both aggregated metrics are rather small.

We cannot compare the results obtained in our experiments directly with the measures based on the graph edit distance and on causal footprints presented in [2]. Due to instantiation issues, we had to exclude 15 models from our experiment, one being a query model. We aim at a direct comparison in future work.

6 Related Work

We elaborated on similarity for process models in Section 2.2. The measure presented in [22] needs further discussion against the background of our metric. The measure is based on behavioral relations that are derived from the tree-structure of a BPEL process. The Jaccard coefficient over all these relations is used as a similarity metric. The relations of [22] virtually coincide with the relations of the behavioral profile used in our metrics. Still, our approach is a generalization of the metric presented in [22]. We (1) leverage the generic concept of a behavioral profile that is independent of a process description language, (2) provide more fine granular metrics that take the interplay of the different relations into account, and (3) proved metric properties.

One of the foremost applications of process similarity that has been nominated by their authors is undoubtedly search in process model collections. Numerous approaches to advanced query languages for process models exist, e.g., [37,38,39], but neither of them addresses efficient search within large collections of process models. Recent approaches to efficient process model search apply a two phase approach: An index is used in a first phase to identify candidate models that could match a given query model, whereas in the second phase the final result set

and ranking is established through exhaustive comparison within the candidate set. Jin et al. [40] propose for exact search of process subgraphs a path-based index, i.e., a hash-based index over concatenated labels of process traces to speed up the first phase. Yan et al. [19] leverage process features, i.e., characteristic structures, such as sequences, splits, and joins, to narrow the candidate list in the first phase of similarity search. To our knowledge, we are the first to apply the metric space approach to behavioral process model similarity that enables efficient search without exhaustive comparison of the data set at all. We showed earlier that metrics can be successfully used to implement efficient similarity search for process model structure [28].

Finally, behavioral profiles have been used to judge the quality of process model alignments, i.e., the quality of correspondences between the activities of two process models [32,4]. Here, preservation of the behavioral relations for corresponding pairs of activities in two process models is quantified. Although this measure may be utilized to also assess process model similarity, it would have several drawbacks. It does not quantify the size of the overlap in terms of shared activities of two process models, which is counter-intuitive for process model similarity search. In addition, the existing measures are no metrics.

7   Conclusion

We motivated our work with efficient similarity search for process models that requires metrics. In this paper, we introduced such a proper metric to quantify behavioral similarity of process models. This metric is built from five elementary similarity measures that are based on behavioral profiles and the Jaccard coefficient. We evaluated the metric experimentally towards its approximation of human similarity assessment.

For our evaluation, we focused on the appropriateness of the metric with respect to human similarity perception instead of its application for similarity search. Our results indicate that the metric is well-suited for assessing the similarity of process models, even though we apply a behavioral abstraction. Further, we already showed that similarity search based on a metric scales well with process model structures and saves up to 80% of comparison operations compared to exhaustive search [28]. Similar results can be expected with the metric proposed in this paper.

In future work, we aim at comparing the proposed metric with the one based on the graph edit distance [17,2] in detail. We also want to investigate the combination of both metrics and their application to more heterogeneous model collections. This may give further insights on how to tune the aggregation weights of our metric.

Finally, activity matching has been out of scope of this paper. We established such a mapping based on the syntactic similarity of labels before calculating the behavioral similarity. The influence of other techniques, e.g., the use of thesauri, may lead to a better alignment of similarity, and shall be examined in future experiments.

References

1. Rosemann, M.: Potential pitfalls of process modeling: part B. Business Process Management Journal **12**(3) (2006) 377–384
2. Dijkman, R., Dumas, M., van Dongen, B., Käärik, R., Mendling, J.: Similarity of business process models: Metrics and evaluation. Inf. Syst. **36**(2) (2011) 498–516
3. Zezula, P., Amato, G., Dohnal, V., Batko, M.: Similarity Search: The Metric Space Approach. Springer-Verlag New York, Inc., Secaucus, NJ, USA (2005)
4. Weidlich, M., Mendling, J., Weske, M.: Efficient Consistency Measurement based on Behavioural Profiles of Process Models. IEEE Trans. Softw. Eng. (2011)
5. Kunze, M., Weidlich, M., Weske, M.: m^3 - A Behavioral Similarity Metric for Business Processes. In: ZEUS'11. CEUR-WS. (2011) 89–95
6. Chávez, E., Navarro, G., Baeza-Yates, R., Marroquín, J.L.: Searching in Metric Spaces. ACM Comput. Surv. **33**(3) (2001) 273–321
7. Hjaltason, G.R., Samet, H.: Index-driven similarity search in metric spaces (Survey Article). ACM Trans. Database Syst. **28**(4) (2003) 517–580
8. Dumas, M., García-Bañuelos, L., Dijkman, R.M.: Similarity Search of Business Process Models. IEEE Data Eng. Bull. **32**(3) (2009) 23–28
9. Rahm, E., Bernstein, P.A.: A survey of approaches to automatic schema matching. VLDB J. **10**(4) (2001) 334–350
10. Euzenat, J., Shvaiko, P.: Ontology matching. Springer-Verlag, Heidelberg (DE) (2007)
11. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics Doklady **10**(8) (1966) 707–710
12. Salton, G., Wong, A., Yang, C.S.: A Vector Space Model for Automatic Indexing. Commun. ACM **18**(11) (1975) 613–620
13. Manning, C.D., Schätze, H.: Foundations of Statistical Natural Language Processing. The MIT Press (1999)
14. Miller, G.A.: WordNet: A Lexical Database for English. Commun. ACM **38**(11) (1995) 39–41
15. Bunke, H., Allermann, G.: Inexact graph matching for structural pattern recognition. Pattern Recognition Letters **1**(4) (1983) 245–253
16. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman (1979)
17. Dijkman, R.M., Dumas, M., García-Bañuelos, L.: Graph Matching Algorithms for Business Process Model Similarity Search. In: BPM. Volume 5701 of LNCS, Springer (2009) 48–63
18. Li, C., Reichert, M., Wombacher, A.: On Measuring Process Model Similarity Based on High-Level Change Operations. In: ER. Volume 5231 of LNCS, Springer (2008) 248–264
19. Yan, Z., Dijkman, R.M., Grefen, P.: Fast Business Process Similarity Search with Feature-Based Similarity Estimation. In: OTM '10, Volume 6426 of LNCS, Springer (2010) 60–77
20. Wombacher, A., Rozie, M.: Evaluation of Workflow Similarity Measures in Service Discovery. In: Service Oriented Electronic Commerce. Volume 80 of LNI., GI (2006) 51–71
21. Nejati, S., Sabetzadeh, M., Chechik, M., Easterbrook, S., Zave, P.: Matching and Merging of Statecharts Specifications. In: ICSE '07, Washington, DC, USA, IEEE Computer Society (2007) 54–64

22. Eshuis, R., Grefen, P.: Structural Matching of BPEL Processes. In: ECOWS '07, Washington, DC, USA, IEEE Computer Society (2007) 171–180
23. van Dongen, B., Dijkman, R., Mendling, J.: Measuring Similarity between Business Process Models. In: CAISE '08, Volume 5074 of LNCS, Springer (2008) 450–464
24. Minor, M., Tartakovski, A., Bergmann, R.: Representation and Structure-Based Similarity Assessment for Agile Workflows. In: ICCBR '07 Volume 4626 of LNCS, Springer (2007) 224–238
25. Ehrig, M., Koschmider, A., Oberweis, A.: Measuring Similarity between Semantic Business Process Models. In: APCCM '07 Australian Computer Society, Inc. (2007) 71–80
26. van der Aalst, W., de Medeiros, A.A., Weijters, A.: Process Equivalence: Comparing Two Process Models Based on Observed Behavior. In: BPM Volume 4102 of LNCS, Springer (2006) 129–144
27. Lu, R., Sadiq, S.: On the Discovery of Preferred Work Practice Through Business Process Variants. In: ER' 07, Volume 4801 of LNCS, Springer (2007) 165–180
28. Kunze, M., Weske, M.: Metric Trees for Efficient Similarity Search in Process Model Repositories. In: IW-PL '10, Hoboken, NJ (September 2010)
29. Vanhatalo, J., Völzer, H., Leymann, F., Moser, S.: Automatic Workflow Graph Refactoring and Completion. In: ICSOC. Volume 5364 of LNCS, Springer (2008) 100–115
30. Lohmann, N., Verbeek, E., Dijkman, R.M.: Petri Net Transformations for Business Processes - A Survey. T. Petri Nets and Other Models of Concurrency **2** (2009) 46–63
31. van der Aalst, W.: Workflow Verification: Finding Control-Flow Errors Using Petri-Net-Based Techniques. In: BPM '09, Volume 1806 of LNCS, Springer (2000) 161–183
32. Weidlich, M., Polyvyanyy, A., Mendling, J., Weske, M.: Efficient Computation of Causal Behavioural Profiles Using Structural Decomposition. In:Petri Nets. Volume 6128 of LNCS, Springer (2010) 63–83
33. Weidlich, M., Mendling, J.: Perceived Consistency between Process Models. Inf. Syst. (2011) In Press.
34. Lipkus, A.: A Proof of the Triangle Inequality for the Tanimoto Distance. Journal of Mathematical Chemistry **26** (1999) 263–265
35. Decker, G., Mendling, J.: Process instantiation. Data Knowl. Eng. **68**(9) (2009) 777–792
36. Buckley, C., Voorhees, E.M.: Evaluating evaluation measure stability. In: SIGIR '00. (2000) 33–40
37. Awad, A.: BPMN-Q: A Language to Query Business Processes. In: EMISA. Volume P-119 of LNI., GI (2007) 115–128
38. Beeri, C., Eyal, A., Kamenkovich, S., Milo, T.: Querying Business Processes. In: VLDB '06VLDB Endowment (2006) 343–354
39. Choi, I., Kim, K., Jang, M.: An XML-based process repository and process query language for integrated process management. Knowledge and Process Management **14**(4) (2007) 303–316
40. Jin, T., Wang, J., Wu, N., Rosa, M.L., ter Hofstede, A.H.M.: Efficient and Accurate Retrieval of Business Process Models through Indexing - (Short Paper). In: OTM '10, Volume 6426 of LNCS, Springer (2010) 402–409