When Process Mining Meets Bioinformatics

R.P. Jagadeesh Chandra Bose[1,2] and Wil M.P. van der Aalst[1]

[1] Department of Mathematics and Computer Science, University of Technology, Eindhoven, The Netherlands
{j.c.b.rantham.prabhakara,w.m.p.v.d.aalst}@tue.nl
[2] Philips Healthcare, Veenpluis 5–6, Best, The Netherlands

Abstract. Process mining techniques can be used to extract non-trivial process-related knowledge and thus generate interesting insights from event logs. Similarly, bioinformatics aims at increasing the understanding of biological processes through the analysis of information associated with biological molecules. Techniques developed in both disciplines can benefit from one another, e.g., sequence analysis is a fundamental aspect in both process mining and bioinformatics. In this paper, we draw a parallel between bioinformatics and process mining. In particular, we present some initial success stories that demonstrate that the emerging process mining discipline can benefit from techniques developed for bioinformatics.

Key words: sequence, trace, execution patterns, diagnostics, conformance, alignment, configuration

1 Introduction

Bioinformatics aims at increasing the understanding of biological processes and entails the application of computational techniques to understand and organize the information associated with biological macromolecules [1]. Sequence analysis or sequence informatics is a core aspect of bioinformatics that is concerned with the analysis of DNA/protein sequences[1] and has been an active area of research for over four decades.

Process mining is a relatively young research discipline aimed at discovering, monitoring and improving real processes by extracting knowledge from event logs readily available in today's information systems [2]. Business processes leave trails in a variety of data sources (e.g., audit trails, databases, and transaction logs). Hence, every process instance can be described by a trace, i.e., a sequence of events. Process mining techniques are able to extract knowledge from such traces and provide a welcome extension to the repertoire of business process

[1] DNA stores information in the form of the base nucleotide sequence, which is a string of four letters (A, T, G and C) while protein sequences are sequences defined over twenty amino acids and are the fundamental determinants of biological structure and function.

analysis techniques. The topics in process mining can be broadly classified into three categories (i) *discovery*, (ii) *conformance*, and (iii) *enhancement*. Process discovery deals with the discovery of models from event logs. For example, there are dozens of techniques that automatically construct process models (e.g., Petri nets or BPMN models) from event logs [2]. Discovery is not restricted to control-flow; one may also discover organizational models, etc. Conformance deals with comparing an apriori model with the observed behavior as recorded in the log and aims at detecting inconsistencies/deviations between a process model and its corresponding execution log. In other words, it checks for any violation between *what was expected to happen* and *what actually happened*. Enhancement deals with extending or improving an existing model based on information about the process execution in an event log. For example, annotating a process model with performance data to show bottlenecks, throughput times etc.

Despite several success stories there are still significant challenges that need to be addressed in applying process mining techniques on real-life event logs. Some of these challenges include:

- *Dealing with less structured processes:* most processes mined from real-life logs tend to be less structured than what stakeholders expect. The discovered process models are often spaghetti-like and are hard to comprehend. Many factors lead to such a behavior e.g., heterogeneity of cases, fine granular events, etc. Process models can be seen as "maps" describing the operational processes of organizations. There is a need for techniques that enable the discovery of *navigable* process maps with seamless zoom-in/zoom-out facility (hierarchical process models with different perspectives).

- *Dealing with fine granular event logs:* some event logs (especially those that emanate from large scale processes, high-tech systems such as medical systems, copiers and scanners, etc) contain events at a very low abstraction level. Stakeholders would like to view processes at a more coarse-grained level. There is a need for (semi-)automated means of aggregating low-level events into high-level events. Voluminous data is a natural side effect of such fine granular event logs. This imposes an additional requirement on the process mining techniques to be scalable as well.

- *Provisions for process diagnostics:* The lion's share of process mining research has been devoted to control-flow discovery. Process diagnostics, which encompasses process conformance checking, auditing, process performance analysis, anomaly detection, diagnosis, inspection of interesting patterns and the like, is gaining prominence in recent years [3, 4, 5, 6, 7]. There is a need for techniques that assist auditors and analysts in their diagnostic efforts [8].

- *Dealing with process changes:* contemporary process mining techniques assume the processes to be in steady state. However, in reality, processes may change to adapt to changing circumstances, e.g., new legislation, extreme variations

in supply and demand, seasonal effects, etc. *Concept drift* refers to the situation in which the process is changing while being analyzed [9]. There is a need for techniques that deal with such "second order dynamics". Analyzing such changes is of utmost importance to get an accurate insight on process executions at any instant of time.

It is important to note that, to a large extent, sequence analysis is a fundamental aspect in almost all facets of process mining and bioinformatics. In spite of all the peculiarities specific to business processes and process mining, the relatively young field of process mining should, in our view, take account of the conceptual foundations, practical experiences, and analysis tools developed by sequence informatics researchers over the last couple of decades. In this paper, we describe some of the analogies between the problems studied in both disciplines. We present some initial successes which demonstrate that process mining techniques can benefit from such a cross-fertilization.

The remainder of this paper is organized as follows. Section 2 introduces some of the basic process mining concepts and illustrates some of the challenges already mentioned. The subsequent sections relate ideas and techniques from bioinformatics to process mining. Section 3 points out similarities in the structuring mechanisms used in both domains, e.g., the hierarchy of protein structures is compared to the hierarchical structuring of events in processes. Section 4 discusses commonalities between alignments in biology and traces in event logs. Section 5 relates phylogeny (the creation of tree structures showing inferred evolutionary relationships among various biological species) to process configuration. Section 6 concludes the paper.

2 Preliminaries: Process Mining

The goal of this paper is to show that process mining can benefit from ideas and techniques originating from bioinformatics. However, before doing so, we first introduce some of the basic process mining concepts and illustrate that there are indeed several problems to be tackled.

Process mining serves a bridge between data mining and business process modeling. The goal is to extract process-related knowledge from event data recorded by a variety of systems (ranging from sensor networks to enterprise information systems). Starting point for process mining is an event log. We assume that events can be related to process instances (often called cases) and are described by some activity name. The events within a process instance are ordered. Therefore, a process instance is often represented as a trace over a set of activities. In real-life event logs, events have timestamps, associated resources (e.g. the person executing the activity), transactional information (e.g., start, complete, or suspend), data attributes (e.g., amount or type of customer). However, for clarity, we abstract from such additional information. Therefore, we can use the following basic notations:

- Σ denotes the set of *activities*. Σ^+ is the set of all non-empty finite sequences of activities from Σ.
- A *process instance* (i.e. case) is described as a *trace* over Σ, i.e., a finite sequence of activities. Examples of traces are `abcd` and `abbbad`.
- Let $T = T(1)T(2)T(3)\ldots T(n) \in \Sigma^+$ be a trace over Σ. $T(k)$ represents the k^{th} activity in the trace. $|T| = n$ denotes the *length* of the trace T.
- An *event log*, $\mathcal{L}$, corresponds to a multi-set (or bag) of traces from Σ^+. For example, $\mathcal{L} = [\texttt{abcd}, \texttt{abcd}, \texttt{abbbad}]$ is a log consisting of three cases. Two cases follow trace `abcd` and one case follows trace `abbbad`.

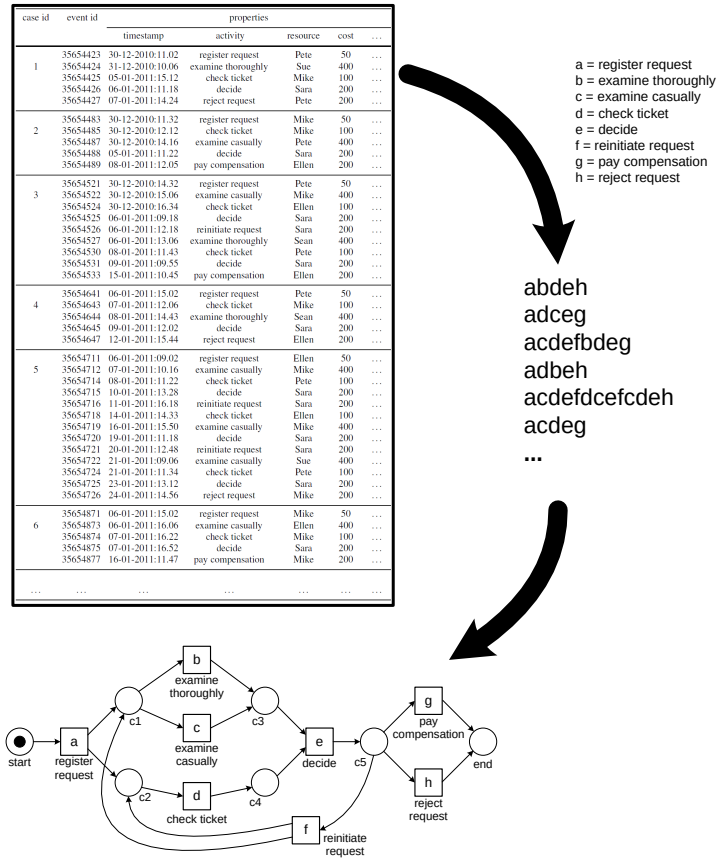


Fig. 1. Process discovery aims to learn a process model (in this case a Petri net) from traces of activities.

As mentioned in Section 1, event logs can be used to conduct three types of process mining: (i) discovery, (ii) conformance, and (iii) enhancement [2]. Process discovery—discovering a process model from example behavior recorded

in an event log—is one of the most challenging tasks in process mining. Today there are dozens of process discovery techniques generating process models using different notations (Petri nets, EPCs, BPMN, heuristic nets, etc.). Fig. 1 illustrates the basic idea of process discovery. An event log containing detailed information about events is transformed into a multiset of traces $\mathcal{L} = [\texttt{abdeh}, \texttt{adceg}, \texttt{acdefbdeg}, \texttt{adbeh}, \texttt{acdefdcefcdeh}, \texttt{acdeg}, \ldots]$. Process discovery techniques are able to discover process models such as the Petri net shown in Fig. 1.

Event logs may be *incomplete* and contain *noise*. Noise refers to rare and infrequent behavior not representative for the typical behavior of the process. Incompleteness refers to the problem that one typically sees only a fraction of all possible behaviors. Traces that are not seen in the log are not necessarily impossible; we only see positive examples and no negative examples. Process mining algorithms need to be able to deal with noise and incompleteness. Generally, we use four main quality dimensions for judging the quality of the discovered process model: *fitness*, *simplicity*, *precision*, and *generalization* [2]. A model with good fitness allows for the behavior seen in the event log. The simplest model that can explain the behavior seen in the log, is the best model (Occam's Razor). A model that is not precise is "underfitting". Underfitting is the problem that the model over-generalizes the example behavior in the log, i.e., the model allows for behaviors very different from what was seen in the log. A model that does not generalize is "overfitting". Overfitting is the problem that a very specific model is generated whereas it is obvious that the log only holds example behavior, i.e., the model explains the particular sample log, but a next sample log of the same process may produce a completely different process model.

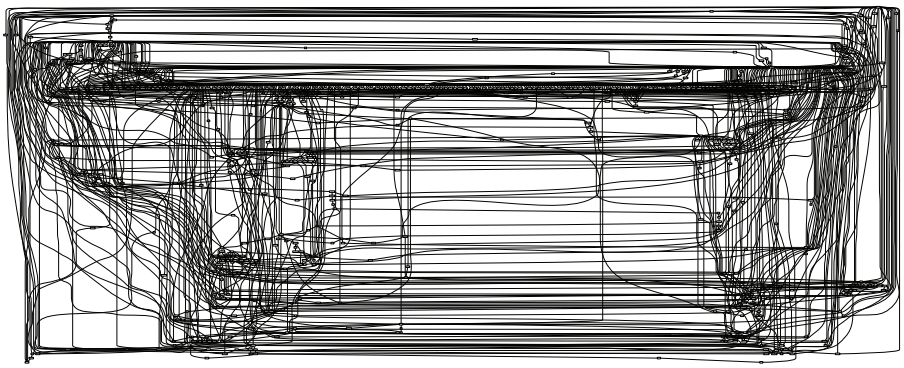


Fig. 2. Spaghetti process describing the diagnosis and treatment of 2765 patients in a Dutch hospital. The process model was constructed based on an event log containing 114,592 events. There are 619 different activities (taking event types into account) executed by 266 different individuals (doctors, nurses, etc.)

The challenges related to process mining are best explained using an example. Fig. 2 shows an example of a typical Spaghetti process discovered using conventional process mining techniques [2]. The complexity of the diagram illustrates the problems and challenges mentioned in Section 1. In the remainder of the paper, we show how ideas and techniques originating from bioinformatics can help to address these.

3 From Sequence to Structure

A DNA *sequence motif* is defined as a nucleic acid *sequence pattern* that has some biological significance (both structural and functional) [10]. These motifs are usually found to recur in different genes or within a single gene. For example, *tandem repeats* (tandemly repeating DNA) are associated with various regulatory mechanisms such as protein binding [11]. More often than not, sequence motifs are also associated with *structural motifs* found in proteins thus establishing a strong correspondence between sequence and structure. Protein structures manifest as a hierarchy of four levels: primary, secondary, tertiary, and quaternary. Primary structure is the basic level and corresponds to the linear sequence of amino acids. Secondary structures result from the regular folding of regions within the amino acid sequence into particular structural patterns e.g., α-helix, β-sheets, β-turns, loops, etc. Tertiary and quaternary structures result from the folding of primary structure and secondary structural elements in 3 dimensions. Fig. 3 depicts the hierarchy of protein structures.

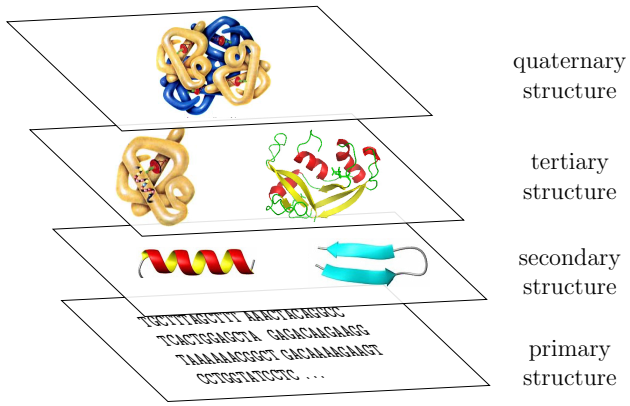


Fig. 3. Hierarchy of protein structures.

Likewise, common subsequences of activities in an event log that are found to recur within a process instance or across process instances have some domain

(functional) significance. In [12], we adopted the sequence patterns (e.g., tandem repeats, maximal repeats etc.) proposed in the bioinformatics literature, correlated them to commonly used process model constructs (e.g., tandem repeats and tandem arrays correspond to simple loop constructs), and proposed a means to form abstractions over these patterns. The abstractions thus uncovered have a strong domain significance from a functionality point of view. Using these abstractions as a basis, we proposed a *two-phase approach to process discovery* [13]. The first phase comprises of pre-processing the event log with abstractions at a desired level of granularity and the second phase deals with discovering the *process maps* with seamless zoom-in/out facility. Fig. 4 summarizes the overall approach. Note the similarity with Fig. 3.

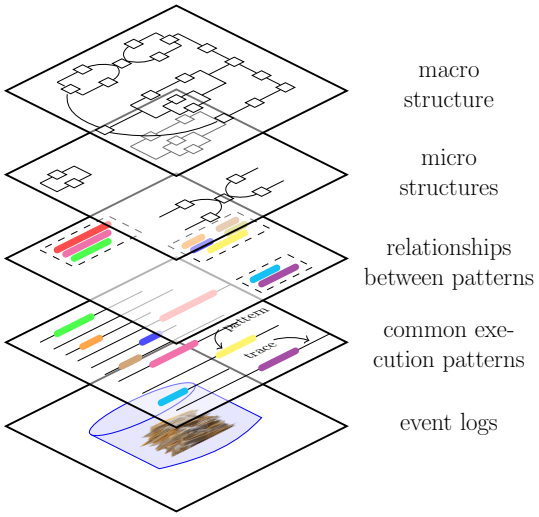


Fig. 4. Repeating subsequences of activities define the common execution patterns and carry some domain (functional) significance. Related patterns and activities pertaining to these patterns define abstractions that correspond to micro-structures (or sub-processes). The top-level process model can be viewed as a macro-structure that subsumes the micro-structures.

Fig. 5 highlights the difference between the traditional approach to process discovery and the two-phase approach. Note that the process model (map) discovered using the two-phase approach is simpler. Our approach supports the abstraction of activities based on their context and type, and provides a seamless zoom-in and zoom-out functionality. Fig. 5 illustrates that a cross-fertilization between bioinformatics and process mining enables the discovery of hierarchical process models. This provides a new perspective when dealing with fine granular event logs and less structured processes.

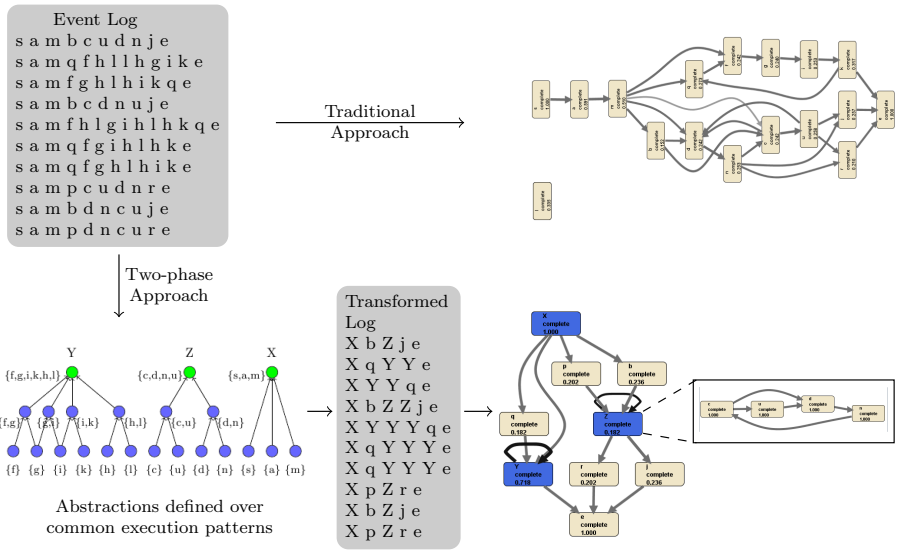


Fig. 5. Traditional approach vs. our two-phase approach to process discovery

4 Sequence Alignment and Process Diagnostics

Multiple sequence alignment has been a subject of extensive research in computational biology for over three decades. Sequence alignment is an essential tool in bioinformatics that assists in unraveling the secondary and tertiary structures of proteins and molecules, their evolution and functions, and in inferring the taxonomic, phylogenetic or cladistic relationships between organisms, diagnoses of genetic diseases, etc. [14, 15].

In [16], we have adapted sequence alignment to traces in an event log and showed that it carries significant promise in process diagnostics. The goal of *trace alignment* is to align traces in such a way that event logs can be easily explored. Given a multi-set of traces $\mathbb{T} = [T_1, T_2, \ldots, T_n]$, trace alignment can be defined as a mapping of $\mathbb{T}$ to another multi-set of traces $\overline{\mathbb{T}} = [\overline{T_1}, \overline{T_2}, \ldots, \overline{T_n}]$ where $\overline{T_i} \in (\Sigma \cup \{-\})^+$ for $1 \leq i \leq n$. In addition, the following three properties need to be satisfied with respect to $\mathbb{T}$ and $\overline{\mathbb{T}}$:

- each trace in $\overline{\mathbb{T}}$ is of the same length i.e., there exists an $m \in \mathbb{N}$ such that $|\overline{T_1}| = |\overline{T_2}| = \cdots = |\overline{T_n}| = m$
- $\overline{T_i}$ is equal to T_i after removing all gap symbols '$-$' and
- there is no $k \in \{1, \ldots, m\}$ such that $\forall_{1 \leq i \leq n}\ \overline{T_i}(k) = -$.

Trace alignment can be used to explore the process in the early stages of analysis and to answer specific questions in later stages of analysis. Fig. 6 depicts

the results of trace alignment for a real-life log from a rental agency[2]. Every row corresponds to a process instance and time increases from left to right. The horizontal position is based on *logical time* rather than real timestamps. If two rows have the same activity name in the same column, then the corresponding two events are very similar and are therefore aligned. Note that the same activity can appear in multiple columns. By reading a row from left to right, we can see the sequence of activities (i.e., the trace) that was executed for a process instance. Process instances having the same trace can be grouped into one row to simplify the diagram. The challenge is to find an alignment that is as simple and informative as possible. For example, the number of columns and gaps should be minimized while having as much consensus as possible per column.

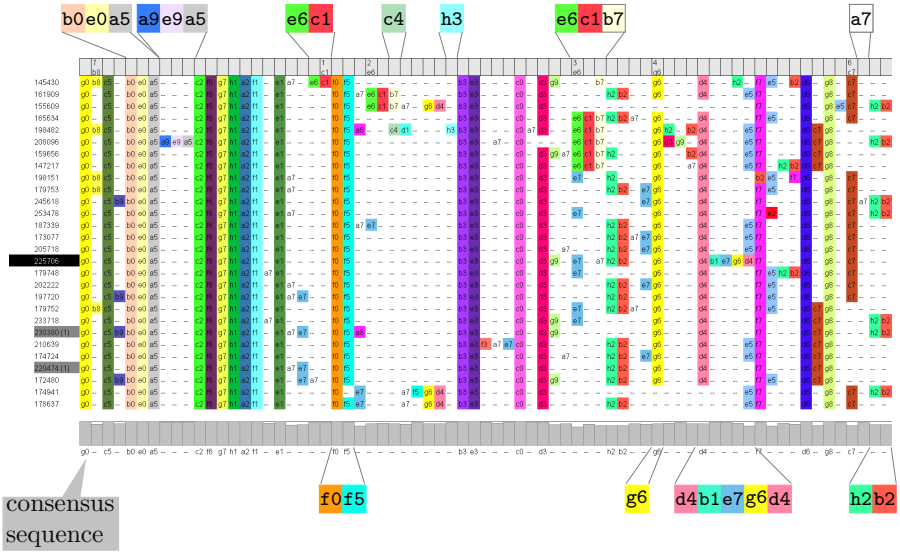


Fig. 6. An example of trace alignment for a real-life log from a rental agency. Each row refers to a process instance. Columns describe positions in traces. Consider now the cell in row y and column x. If the cell contains an activity name `a`, then `a` occurred for case y at position x. If the cell contains no activity name (i.e., a gap "−"), then nothing happened for y at position x.

Trace alignment can assist in answering a variety of diagnostic questions. For example, one can get answers to questions such as:

– *What is the most common (likely) process behavior that is executed?*
The consensus sequence of an alignment, which captures the major activity in each column, represents the most common process behavior that is executed

[2] Since the whole alignment is not legible, we highlight the interesting patterns/activities at the top and the bottom of the figure.

and can be considered as the back-bone sequence for the process.

- *Are there any common patterns of execution in my traces?*
  Common execution patterns are captured in the form of well conserved regions (columns) in the alignment. For example, the activity sequence `b0e0a5` (at columns $5-7$) corresponding to the activities, `planning of first inspection`, `preparation of lease termination form`, and `is first inspection performed?` respectively, is common across all the traces.

- *Where do my process instances deviate and what do they have in common?*
  Deviations, exceptional behavior and rare event executions are captured in regions that are sparsely filled i.e., regions with lot of gap symbols ($-$) or in regions that are well conserved with a few rare gaps.
  For example, it could be seen that only one of the traces (sixth trace in the alignment) has the activity subsequence `a9e9a5` in columns $8-10$. Activity `a5` in column 7 corresponds to the check, `is first inspection performed?` and the activity subsequence `a9e9a5` corresponds to the scenario where the result of the check was negative due to the fact that the tenant was not at home. `a9` corresponds to the activity of sending a letter to the tenant and `e9` corresponds to the activity of rescheduling the first inspection.

- *What are the contexts in which an activity or a set of activities is executed in my event log?*
  Trace alignment provides a complete perspective of activity executions in a log including that of long range dependencies (any dependencies between activities are reflected as common execution patterns in the traces where they manifest). Furthermore, with rich interactive visualization (such as the options of filtering columns containing an activity), trace alignment enables a flexible inspection of the log.

- *What are the process instances that share/capture a desired behavior either exactly or approximately?*
  One can formulate the desired behavior as an activity sequence and apply trace alignment of this sequence with the traces in the log. Traces/process instances that share the desired behavior have a lot of their activities aligned with that of the activities in the desired behavior sequence.

- *Are there particular patterns (e.g., milestones, concurrent activities etc.) in my process?*
  Concurrent activities manifest in mutually exclusive traces across different columns in an alignment. For example, the activities `h2b2` corresponding to the `drafting of final note` (`h2`) and `archiving of lease termination` (`b2`) is concurrent in this process.

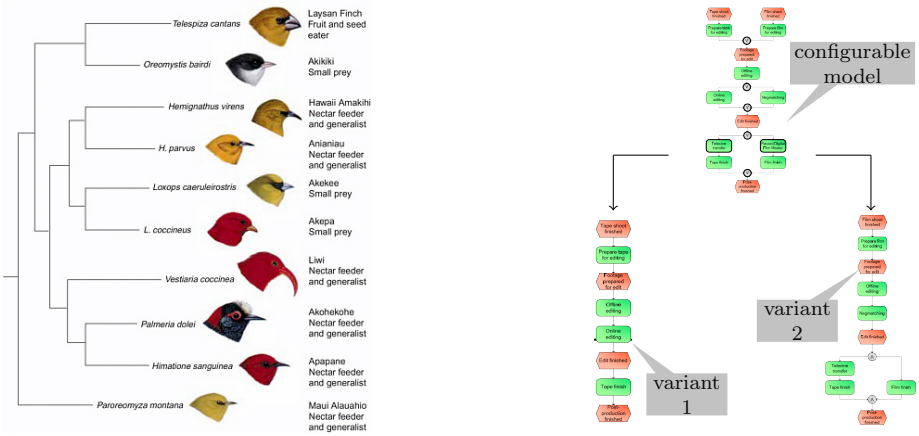
The application of sequence alignment in bioinformatics to process mining has created an altogether new dimension to conformance checking; *deviations and*

violations are uncovered by analyzing just the raw event traces (thereby avoiding the need for process models).

Finding good quality alignments is notoriously complex. The initial results of trace alignment are definitely encouraging. Nonetheless, there are various new challenges when adopting biological sequence alignment to trace alignment in the context of business processes [17]. For example, biological sequences tend to be homogenous whereas traces in semi-structured processes (e.g., care processes in hospitals) tend to be heterogeneous. Other differences are the fact that traces in an event log can be of very different lengths (e.g., due to loops) and may be the result of concurrency. These characteristics provide new challenges for sequence alignment.

5 Phylogeny and Process Configuration

Phylogenetics refers to the study of evolutionary relationships, and was one of the first applications in bioinformatics. A phylogeny is a tree representation of the evolutionary history of a set (family) of organisms, gene/protein sequences etc. The basic premise in phylogenetics is that genes have evolved by duplication and divergence from common ancestors [18]. The genes can therefore exist in a nested hierarchy of relatedness. Fig. 7(a) depicts the phylogeny of some of the species of Hawaiian honeycreeper [19]. These variant species descended from a single species over the last ten million years.



(a) Phylogeny of the Hawaiian honeycreeper.

(b) Process configuration

Fig. 7. Similarity between phylogeny and process configuration.

Phylogeny is related to structuring variability within and between processes. In the past couple of years, *process configuration* has gained prominence in the BPM community [20]. Process configuration is primarily concerned with managing families of business processes that are similar to one another in many ways yet differing in some other ways. For example, processes within different municipalities are very similar in many aspects and differ in some other aspects. Such discrepancies can arise due to characteristics peculiar to each municipality (e.g., differences in size, demographics, problems, and policies) that need to be maintained. Furthermore, operational processes need to change to adapt to changing circumstances, e.g., new legislation, extreme variations in supply and demand, seasonal effects, etc. A configurable process model describes a family of similar process models in a given domain [20], and can be thought of as the genesis (root) of the family. All variants in the family can be derived from the configurable model through a series of change patterns [21]. Fig. 7(b) depicts an example of a configurable model (parent) and two variants (children) derived from it. One of the core research problems in *process configuration* is to automatically derive configurable process models from specific models and event logs.

One can find stark similarity between phylogenetics and process configuration. Techniques have been proposed in the bioinformatics literature to discover phylogenies both from (protein) structure as well as from sequences. This can be compared to deriving configurable process models from specific models and from event logs respectively. The adaptability of phylogeny construction techniques to process configuration needs to be explored.

Techniques from bioinformatics have also been adopted to trace clustering in process mining [22, 23]. Trace clustering was shown to be effective in dealing with the heterogeneity in event logs [22, 23]. Process mining results can be improved by segregating heterogeneous cases into more homogenous clusters and analyzing each cluster separately. Sequence clustering techniques have been applied to deal with unlabeled event logs[3] in process mining [24]. Experiences from bioinformatics can also contribute to tooling and infrastructure efforts in process mining. For example, visualization is one of the challenging problems in process mining tooling[4]. A lot of current visualization means in process mining become unmanageable when dealing with large event logs thereby compromising the comprehensibility. Process mining is typically an iterative activity driven by questions from stakeholders and surprising analysis results. Techniques for visualization in process mining should focus on supporting the strong iterative and interactive nature of event log analysis e.g., ranging from overview results to focused and directed insights, annotating mined results, enabling holistic views by juxtaposing several different analysis results simultaneously, etc. *Visualization*

[3] In an unlabeled event log, the case to which an event belongs to is unknown.

[4] ProM is an extensible framework that provides a comprehensive set of tools/plugins for the discovery and analysis of process models from event logs. See `http://www.processmining.org` for more information and to download ProM.

is used in many areas within bioinformatics (e.g., sequence matching, genome browsing, multiple sequence alignment, etc.), with varying success, and good tools already exist. There is significant potential to learn from the success stories that bioinformatics reveal, e.g., event logs refer to multi-sets of traces, which are basically collections of sequences; sequence exploration and visualization techniques in bioinformatics can be assessed for their adoption to event logs.

Benchmarking and data repositories form another area where bioinformatics has matured over the years. To cater to the rapidly increasing accumulation of biological data, lots of efforts had been initiated in bioinformatics to create advanced databases with analysis capabilities devoted to particular categories e.g., Genbank (cataloguing DNA data), SWISS-PROT/TrEMBL (repository of protein sequences), etc. These repositories support features such as protein sequence/structural/functional comparison and classification benchmarks. Process mining being an emerging technology, such repositories and good benchmarks are still missing. Recently, several efforts had been initiated in the process modeling and process mining community to create repositories with advanced support for dealing with process model collections e.g., APROMORE [25], and repositories of event logs [26]. Process mining repositories and benchmarks should include:

- event logs and process mining tasks e.g., control-flow discovery, organizational model extraction, etc.
- event logs, process models and associated tasks e.g., process conformance, replay techniques, etc.
- process models with associated characteristics e.g., functional (such as loan application process), structural (such as the workflow patterns present), behavioral, etc.

Event log and process model comparison methods, search, and exploration are some of the essential features that these repositories need to support. Quality metrics (e.g., fitness, precision, generalization, computational complexity, etc.) of state-of-the-art techniques also need to be captured in these repositories. This enables the comparison of performance of a new algorithm/technique with contemporary methods. It is also desirable to elicit validation protocols to streamline the ways in which such quality metrics are measured.

Such an overlap between the goals combined with the promising initial results calls for a more rigorous attempt at understanding and exploiting the synergy between these two disciplines.

6 Conclusions

Bioinformatics and process mining share some common goals. In this paper, we presented the commonalities between the problems and techniques studied in bioinformatics and process mining. Exploiting these commonalities, we demonstrated that process mining can benefit from the plethora of techniques developed

in bioinformatics. Initial attempts at such a crossover have enabled the discovery of hierarchical process models and helped extending the scope of conformance checking to also cover the direct inspection of traces. Although this is just a first step towards an interaction between the two disciplines, the results are very promising and the relationship will be explored further in our future work.

Acknowledgments The authors are grateful to Philips Healthcare for funding the research in process mining.

References

1. Luscombe, N., Greenbaum, D., Gerstein, M.: What is Bioinformatics? A Proposed Definition and Overview of the Field. Methods of Information in Medicine **40**(4) (2001) 346–358
2. van der Aalst, W.M.P.: Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer (2011)
3. Rozinat, A., van der Aalst, W.M.P.: Conformance Checking of Processes Based on Monitoring Real Behavior. Information Systems **33**(1) (2008) 64–95
4. van der Aalst, W.M.P., van hee, K.M., van der Werf, J.M., Verdonk, M.: Auditing 2.0: Using Process Mining to Support Tomorrow's Auditor. Computer **43**(3) (2010) 90–93
5. van der Aalst, W.M.P., de Medeiros, A.K.A.: Process Mining and Security: Detecting Anamolous Process Executions and Checking Process Conformance. Electronic Notes in Theoretical Computer Science **121** (2005) 3–21
6. Yang, W.S., Hwang, S.Y.: A Process Mining Framework for the Detection of Healthcare Fraud and Abuse. Expert Systems with Applications **31**(1) (2006) 56–68
7. Bezerra, F., Wainer, J., van der Aalst, W.M.P.: Anomaly Detection Using Process Mining. In: Enterprise, Business-Process and Information Systems Modeling. Volume 29 of LNBIP. Springer (2009) 149–161
8. van der Aalst, W.M.P.: Challenges in Business Process Mining. Technical Report BPM-10-01, Business Process Management (BPM) Center (2010)
9. Bose, R.P.J.C., van der Aalst, W.M.P., Žliobaitė, I., Pechenizkiy, M.: Handling Concept Drift in Process Mining. In: Proceedings of the 23rd International Conference on Advanced Information Systems Engineering (CAiSE). Volume 6741 of LNCS., Springer (2011) 391–405
10. Das, M.K., Dai, H.K.: A Survey of DNA Motif Finding Algorithms. BMC Bioinformatics **8**(Suppl 7) (2007) S21
11. Kolpakov, R., Bana, G., Kucherov, G.: mreps: Efficient and Flexible Detection of Tandem Repeats in DNA. Nucleic Acids Research **31**(13) (2003) 3672–3678
12. Bose, R.P.J.C., van der Aalst, W.M.P.: Abstractions in Process Mining: A Taxonomy of Patterns. In Dayal, U., Eder, J., Koehler, J., Reijers, H., eds.: Business Process Management. Volume 5701 of LNCS., Springer-Verlag (2009) 159–175
13. Li, J., Bose, R.P.J.C., van der Aalst, W.M.P.: Mining Context-Dependent and Interactive Business Process Maps using Execution Patterns. In zur Muehlen, M., Su, J., eds.: BPM 2010 Workshops. Volume 66 of LNBIP., Springer-Verlag (2011) 109–121

14. Chan, S., Wong, A.K.C., Chiu, D.: A Survey of Multiple Sequence Comparison Methods. Bulletin of Mathematical Biology **54**(4) (1992) 563–598
15. Gotoh, O.: Multiple Sequence Alignment: Algorithms and Applications. Advanced Biophysics **36** (1999) 159–206
16. Bose, R.P.J.C., van der Aalst, W.M.P.: Trace Alignment in Process Mining: Opportunities for Process Diagnostics. In Hull, R., Mendling, J., Tai, S., eds.: Proceedings of the 8th International Conference on Business Process Management (BPM). Volume 6336 of LNCS., Springer-Verlag (2010) 227–242
17. Notredame, C.: Recent Progress in Multiple Sequence Alignment: A Survey. Pharmacogenomics **3** (2002) 131–144
18. Thornton, J.W., DeSalle, R.: Gene Family Evolution and Homology: Genomics Meets Phylogenetics. Annual Review of Genomics and Human Genetics **1**(1) (2000) 41–73
19. Olson, S.: Evolution in Hawaii: A Supplement to Teaching About Evolution and the Nature of Science. National Academic Press (2004)
20. van der Aalst, W.M.P., Lohmann, N., Rosa, M.L., Xu, J.: Correctness Ensuring Process Configuration: An Approach Based on Partner Synthesis. In Hull, R., Mendling, J., Tai, S., eds.: Proceedings of the 8th International Conference on Business Process Management (BPM). Volume 6336 of LNCS., Springer-Verlag (2010) 95–111
21. Weber, B., Rinderle, S., Reichert, M.: Change Patterns and Change Support Features in Process-Aware Information Systems. In: Proceedings of the 19th International Conference on Advanced Information Systems Engineering (CAiSE), Springer-Verlag (2007) 574–588
22. Bose, R.P.J.C., van der Aalst, W.M.P.: Context Aware Trace Clustering: Towards Improving Process Mining Results. In: Proceedings of the SIAM International Conference on Data Mining (SDM). (2009) 401–412
23. Bose, R.P.J.C., van der Aalst, W.M.P.: Trace Clustering Based on Conserved Patterns: Towards Achieving Better Process Models. In: Business Process Management Workshops. Volume 43 of LNBIP., Springer (2010) 170–181
24. Ferreira, D., Zacarias, M., Malheiros, M., Ferreira, P.: Approaching Process Mining with Sequence Clustering: Experiments and Findings. In: Proceedings of the 5th International Conference on Business Process Management (BPM). Volume 4714 of LNCS., Springer (2007) 360–374
25. Rosa, M.L., Reijers, H.A., van der Aalst, W.M.P., Dijkman, R.M., Mendling, J., Dumas, M., Garcia-Banuelos, L.: APROMORE: An Advanced Process Model Repository. Expert Systems with Applications **38**(6) (2011) 7029–7040
26. 3TU.DataCentrum: `http://data.3tu.nl/repository/collection:event_logs`.