Chess Mining - Process Models Over The Board

Micha Birklbauer, B.Sc.
University of Applied Sciences Upper Austria
Hagenberg i.M., Austria

Dorian Karaban, B.Sc.
University of Applied Sciences Upper Austria
Hagenberg i.M., Austria

ABSTRACT

Problem: Although chess games do not necessarily have timestamps they are neatly ordered and can be expressed as sequences of moves. A proper sequence of moves is crucial at the beginning of the game. They serve as a starting situation for the further course of the game. Therefore, it is of vital interest to a player how a game develops at the beginning. In the following we suggest modelling chess games using process mining (PM) tools.
Data: Matches from two different players - namely Dorian and Robert Fischer - as well as some popular openings scrapped from various websites are used. Considered attributes are the match number (case ID), the move e.g. “e4” (activity), the colours (resource) and the turn (timestamp) to design an event log.
Goal: The goal is to compare openings - which we hereby define as the first 10 turns - of different users and look for patterns in the generated process model as well as conformance to a standard model and between players.
Results: Modeling chess games as process models for interpretation and comparison is very much possible. In fact one can clearly find openings as defined in chess encyclopedias in the resulting graphs and nets. Furthermore calculating conformance to a standard process also yields reasonable and interpretable results. However findings show that comparing player to player may not be reasonable with process mining techniques.

PVLDB Reference Format:
Micha Birklbauer, B.Sc. and Dorian Karaban, B.Sc.. Chess Mining - Process Models Over The Board. PVLDB, 14(1): XXX-XXX, 2020.
doi:XX.XX/XXX.XX

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/t0xic-m/chess_mining.

1 INTRODUCTION

As a game of chess unfolds one encounters many different variants. Especially towards the end of the game, the possibilities to checkmate the opponent accumulate (for more on solving these chess patterns, see footnote). Experienced chess players, large databases and other factors resulted in chess theories that define good and/or the best move. These theoretical moves can be learned, for example, to get into a predefined chess structure within the first 5-10 moves. From this position, depending on the skill of the two players, the game develops in different directions.

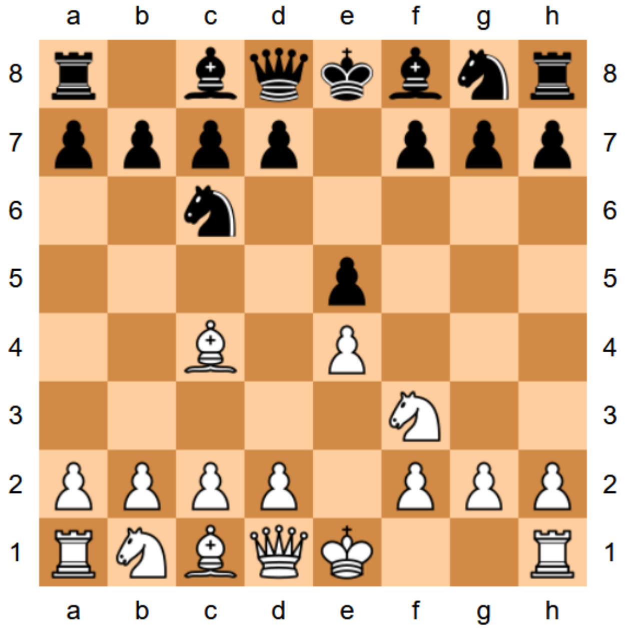

Figure 1: An illustration of a possible chess position following a trace. Picture from Wikipedia.

By means of PM techniques one would like to find out whether or not one can recognize a standardized structure in these processes. Based on this, the following two goals are defined:

(1) Finding or creating "standard processes" that have been selected according to certain criteria that come from domain knowledge (Process Discovery).
(2) Recognizing deviations in future processes by comparison with above mentioned standard processes as well as comparing play styles between different players (Conformance Checking).

In this context it is important that the database is limited to previously defined initial moves to narrow down the scope. Furthermore, one would like to use data produced by exactly one player. The expectation is to get to know play patterns and game behavior of said player. Chess offers a complex domain where process mining techniques can be pushed to their limits to see how well they can perform [4].

2 COLLECT EVENT DATA

Data was scraped in the standard portable game notation (pgn) format from several online chess sites like chess.com and lichess.org. The pgn files were then loaded and preprocessed in python using the package python-chess. The output of the preprocessing pipeline

Proceedings of the VLDB Endowment, Vol. 14, No. 1 ISSN 2150-8097.
doi:XX.XX/XXX.XX

was an eventlog in CSV format which was loaded and converted to XES in Disco [5] [1].

3 METHODS

3.1 Visualization

To visualize tile usage and get a general overview of frequent moves the visualization tool of choice were 8x8 heatmaps resembling chess boards.

3.2 Manual Narrowing

Here the model is manually discovered by analysing the Directly follow graph. In chess terms one could refer to that as the search for the pattern of the "Italian Game".

In the manual analysis, the data set is imported into the DISCO application. Using one of it's functionalities, i.e. the direct-follow graph, the raw dataset is firstly interpreted and later narrowed down for the subsequent model search.

The event log consists of 2129 chess games. All games were played by Grandmasters. The lowest Elo is 1950 and the highest Elo is 2851, resulting in a mean Elo of 2482 across all players. The Elo is a measurement for the skill level according to the international chess federation, short FIDE (*"Fédération Internationale des Échecs"*). All games are described by a code one can find in the chess encyclopedia [2].In this case, the first activities we are looking for are equivalent to the first moves in a chess game. The code determines which specific moves are involved. In this manner a large set of combinations is saved in a uniform encoding. For this purpose we picked the codes C50-59. That corresponds to games of the so-called Giucio Piano or also called "Italian Game". The first five activities to be expected are corresponding with the following trace

$$t_i = \{e4, e5, Nf3, Nc6, Bc4, ...\}.$$

Deviations and alternative combinations are involved in the code C50-59. Some of them differ at the start of the traces, but especially from the sixth timestamp multiple variants can appear. The trace t is encoded with C54. After all, several variations are mapped to the Italian Game via the encoding. The first goal is to discover those patterns in the event log with the directly-follow-graph which are called variations later on. Moreover different miners are used to acknowledge the manual results.

The number of variations in the event log exceeds expectations. To reduce the complexity of the events, a noise is defined. In this work, the noise denotes games that are characterized by a rare variation. It means that a trace differs from almost all of the other Traces, e.g. the trace

$$t_i = \{e4, e5, Nf3, Nc6, Bc4, d6, c3, Be6, Bb5, a6\}.$$

occurs only once in 2129 cases. The maximum frequency for 82% of the variations in the event log is three. Those traces with a rare variation are removed. For this reason the event log is reduced by almost a quarter of the data. A dataset with 1581 cases and 11 variations remains. The most frequent variations look as follows. These sequences of activities describe 87.86% of the already filtered events.

$$\begin{aligned}
v_1(26.63\%) &= \{e4, e5, Nf3, Nc6, Bc4, Bc5, c3, Nf6, d3, d6\}\\
v_2(21.38\%) &= \{e4, e5, Nf3, Nc6, Bc4, Bc5, c3, Nf6, d3, a6\}\\
v_3(16.76\%) &= \{e4, e5, Nf3, Nc6, Bc4, Bc5, c3, Nf6, d4, exd4\}\\
v_4(7.27\%) &= \{e4, e5, Nf3, Nc6, Bc4, Nf6, d3, Bc5, c3, d6\}\\
v_5(6.45\%) &= \{e4, e5, Nf3, Nc6, Bc4, Nf6, d3, Bc5, c3, a6\}\\
v_6(5.76\%) &= \{e4, e5, Nf3, Nc6, Bc4, Bc5, c3, Nf6, b4, Bb6\}\\
v_7(3.61\%) &= \{e4, e5, Nf3, Nc6, Bc4, Bc5, O-O, Nf6, d3, d6\}
\end{aligned}$$

3.3 Standard Model

Discovering a model with process mining algorithms: With the use of the PM4PY library three different mining algorithms are applied to the data. The event log, which was used to create the directly-follow-graph, is used as the data basis. The goal is to find out which miner creates the best model for the log that represents the chess process as it is described in the literature and the chapter before.

For the evaluation of the individual models the resulting graphs are considered. Depending on the miner, three different types of graph are distinguished. The graphs are created with default parameters and can be converted into petri nets which are relevant for the subsequent conformance checks.

1. For the Alpha Miner a petri net is considered.
2. The inductive miner returns an inductive tree by default.
3. Finally, a heuristic net is created with the heuristic miner.

For the analysis a graph is created with each of the three methods. Furthermore, this is used to manually determine by visual interpretation which miner realizes the processes in the event log well. Here it should be noted that it is important that the main process

$$t_i = \{e4, e5, Nf3, Nc6, Bc4, ...\}$$

occurs in the graphs. That sequence is already documented in a game from the 19th century, in which Dubois played against W.Steinitz[3]. However, the model should also be as general as possible to be able to explain deviating variations in the event log as pointed out and explained in the section Manual Narrowing. An alternative process could be

$$v_3(16.76\%) = \{e4, e5, Nf3, Nc6, Bc4, Bc5, c3, Nf6, d4, exd4\}$$

and the very similar trace

$$v_6(5.76\%) = \{e4, e5, Nf3, Nc6, Bc4, Bc5, c3, Nf6, b4, Bb6\}$$

The two traces v_3 and v_6 do not only differ among themselves but compared to the main trace v_1 both also have a different final activity. Such differences should be realized by the model.

3.3.1 *Alpha Miner.*

The petri net that emerged from the Alpha Miner is not well suited for visual interpretation. From the second activity on, the traces start to evolve in different directions, making the model difficult to interpret.

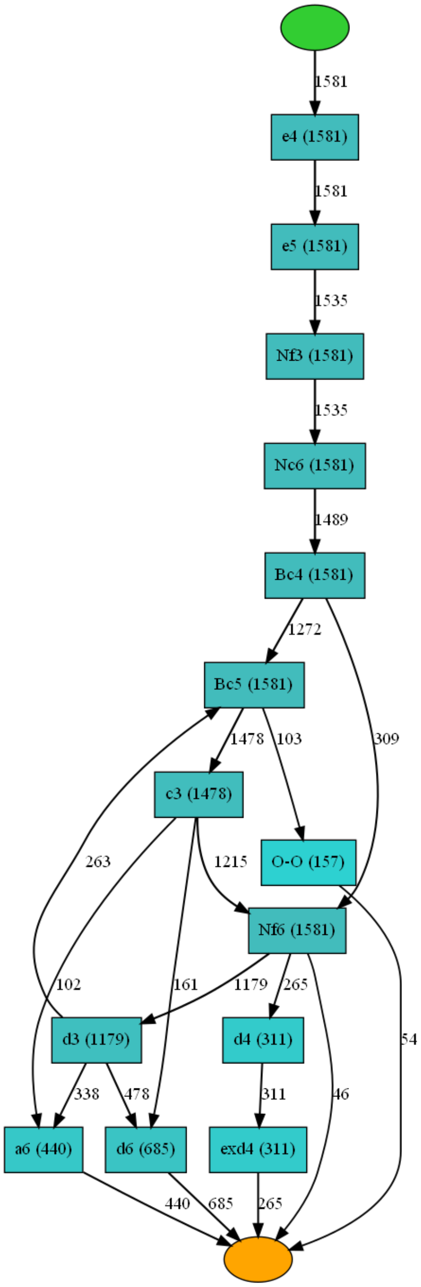

Figure 2: Heuristic Net - Evaluated Standard Model for 1581 Grandmaster Games.

3.3.2 Inductive Miner. In the second approach, the inductive miner was used to create a model which was subsequently compared to the previous one. The resulting petri net is visually divided into three chunks of activities. These three groups are also well recognizable in the inductive tree. they offer an interesting interpretation of the process. The advantage of the inductive miner is that subprocesses can be found very well from the model [6]. The following interpretations can be stated at this point.

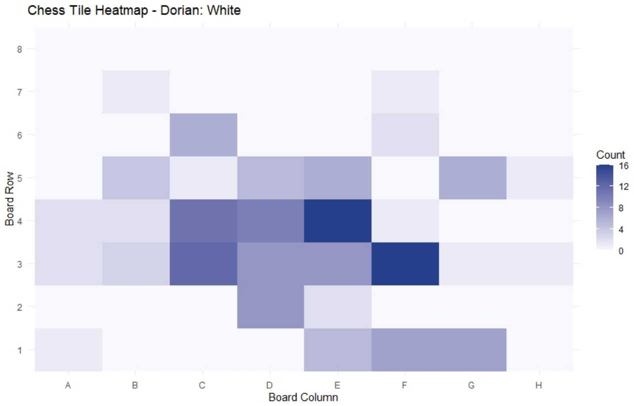

Figure 3: Tile Usage of Dorian.

(1) The start activities are the same as within the alpha miner result.
(2) The traces start to develop a variation in the center. Between the third and sixth, as well as between the seventh and eighth, two sub-trees with their own activities are formed.
(3) The end activities are isolated from the other activities in the trace, i.e. they tend to occur at the end of a trace.

3.3.3 Heuristic Miner. Last but not least a heuristic net was created using the heuristic miner. Like all other graphs, the heuristic net starts with the two starting activities <e4,e5>. However, unlike the other results, this one shows a clear progress of five subsequent activities. In Figure 2 one sees how the model develops only starting from the activity *Bc*4 in different directions. For the first half, this infers that a trace

$$T = \{e4, e5, Nf3, Nc6, Bc4, ...\}$$

is derived. The fact that the trace T represents over 87.86% of the traces has led to the usage of that event log as a standard model.

In the next chapter it is shown how the different petri nets affect the calculation of the fitness. Two methods are used to calculate the fitness. First, all three graphs are converted to a petri net to perform token based replay on them. As a second method, the fitness is calculated by aligning the event data to the standard log. It shows how fitness quality corresponds to the interpretation of the graphs described in this chapter.

3.4 Fischer Model

Moreover, as mentioned in the previous sections the goal was not only to compare games to a defined standard process but also to check for similarities and conformance between players. Henceforth for this purpose 100 games of popular chess Grandmaster Robert Fischer were scraped from chess.com and modeled with an inductive miner. For conformance checking the method of choice was an alignment approach.

Table 1: Fitness Results - Standard Model

Alpha M.	Eventlog (1581)	Small Snapshot (2̃0)	Italian Games (9̃0)
Token based replay Alignement	0.843 0.611	0.629 0.391	0.726 0.477
Inductive M.			
Token based replay Alignement	1.000 1.000	0.691 0.558	0.784 0.786
Heuristics M.			
Token based replay Alignement	0.986 0.973	0.680 0.503	0.856 0.773

Table 2: Conformance Metrics - Dorian VS Fischer Games

Colour	Fitness	Generalization	Precision	Simplicity
White	0.981	0.107	1.000	0.360
Black	0.979	0.112	1.000	0.363

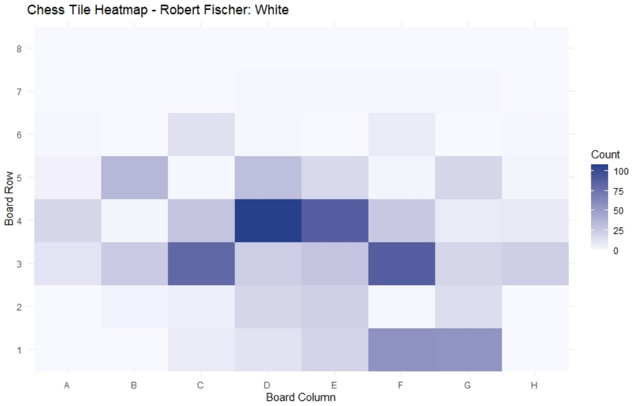

Figure 4: Tile Usage of Robert Fischer.

4 RESULT

4.1 Tile Usage

Tile usage of both analyzed players - namely Dorian and Robert Fischer - can be seen in Figure 3 and Figure 4 respectively. Some noticeable differences are that Dorian prefers "e4" over "d4" while Robert Fischer prefers the other way around.

4.2 Conformance with Standard Model

Two methods are used to calculate the fitness. At the beginning all three resulting graphs were converted to petri nets to perform token based replay. As a second method, the fitness is calculated by aligning the event data to the log. The Table 1 contains final fitness results. Each column represents the fitness value for a specific log file.

(1) Eventlog (1581), log which has been used to create the petri nets.
(2) Small Snapshot (20), a sample of randomly selected games during a period of time.
(3) Italian Games (90), log file is filtered by chess code´s beforehand.

For token based replay, the fitness is defined as the percentage of successfully replayed activities in the log file. Six final values per log files are calculated because each log has been evaluated with every graph and conformance checking method. The quality of the small snapshot does not change very much by using another petri net as the standard reference. We expected to obtain a better fitness if the logs are already filtered by a specific manner, in our case the logs has been filtered by the chess codes. This step improved the fitness which we can see in the column "Italian Games". The statement that the results improve by adjusting the log file remains true.

4.3 Process Model

By restricting the event data and analyzing the corresponding directly follow graph, it was possible to find similar variants described in the chess encyclopedia. The model in Figure 2 shows the sequence found in the literature under the code C50-59 [2].

In order to find that model appropriate assumptions had to be made in the preprocessing stage, which in retrospect turned out to be true. Firstly it should be noted that the incredible variation of chess moves is very high already from the second move. There are 72084 possible positions after each player moves twice. Obviously the quality of the sequences that leads to such position are not equally good. By narrowing down the games to chosen chess openings the scope has been narrowed to a dozen possible positions. The noise mentioned in chapter 3.1 is not taken into account, because otherwise it would not be practical to include that event log in the further methods. By the described procedure a lot of noise was removed. In the context of chess, it is important to mention that the games that were previously declared as noise are actually games that need to be emphasized. Grandmasters develop special strategies how to bypass common theory on purpose in order to eventually checkmate the opponent. In this paper, however, the focus is on discovering the standard openings via process mining rather than teaching high quality Grandmaster games. Jan Pinski describes in his book "The Italian Game and Evans Gambit" the sequence,

$$v_3(16.76\%) = \{e4, e5, Nf3, Nc6, Bc4, Bc5, c3, Nf6, d4, exd4\}$$

as part of the classical "Italian Game". In our model, this sequence counts as the third most frequent variation. In contrast to the trace T that is only half as long

$$T = \{e4, e5, Nf3, Nc6, Bc4, ...\},$$

the opening mentioned by Jan Pinksi maps a complete trace of 10 activities. In our results we can show that the most frequent variations of these traces in the event log can correspond to the move sequences given in the chess encyclopedia. Chess notations can be transformed into an event log with proper preprocessing. The chess logs can be visualized by a directly-follow-graph and cast into models by a appropriate miner.

4.4 Conformance with Fischer Games

If one compares the openings of Dorian to the openings of Robert Fischer with conformance checking as described in section 3.4 it results in the metrics shown in Table 2 Contrary to expectation the values for fitness as well as precision are very high and it opens up the question if that really makes sense. Not necessarily from the number point of view but if doing such a comparison with conformance checking is actually sound and applicable. The 100 Fischer openings may already cover a broad spectrum of moves and from the high fitness values it is apparent that most of the traces from Dorian are covered - even though they might play totally different. We therefore conclude that this comparison does not yield any useful new knowledge in this case.

REFERENCES

[1] 2016. IEEE Standard for eXtensible Event Stream (XES) for Achieving Interoperability in Event Logs and Event Streams. *IEEE Std 1849-2016* (2016), 1–50. https://doi.org/10.1109/IEEESTD.2016.7740858

[2] Sahovski Informator Beograd. 2005. *Encyclopedia of Chess Openings* (5. ed.). Vol. C. Chess Informant, Serbia.

[3] Tony Cullen. 2020. *Chess Rivals of the 19th Century: With 300 Annotated Games*. McFarland & Company, United States of America.

[4] Stefano Ferilli and Sergio Angelastro. 2017. Mining Chess Playing as a Complex Process. In *New Frontiers in Mining Complex Patterns*, Annalisa Appice, Michelangelo Ceci, Corrado Loglisci, Elio Masciari, and Zbigniew W. Raś (Eds.). Springer International Publishing, Cham, 248–262.

[5] Fluxicon, Christian and Anne. [n.d.]. *DISCO*. https://fluxicon.com/disco/

[6] Wil M. P. van der Aalst. 2016. *Process Mining: Data Science in Action* (2 ed.). Springer, Heidelberg. https://doi.org/10.1007/978-3-662-49851-4